EU FP7 CogX
ICT-215181
May 1 2008 (52months)

# DR 3.1:
# Object based representations of space

Patric Jensfelt[1], Alper Aydemir[1], Adrian Bishop[1], Geert-Jan M. Kruijff[2], Andrzej Pronobis[1], Kristoffer Sjöö[1], Hendrik Zender[2]

[1]*Kungliga Tekniska Högskolan, Stockholm*
[2]*DFKI GmbH, Saarbrücken*
⟨patric@kth.se⟩

| | |
|---|---|
| *Due date of deliverable:* | July 31 2010 |
| *Actual submission date:* | July 27 2010 |
| *Lead partner:* | KTH |
| *Revision:* | draft |
| *Dissemination level:* | PU |

---

This is the first deliverable from WP3 and thus represents the work over the first two years in the area of qualitative spatial cognition. One of the main outcomes of the work are the design principles of the overall spatial representation for CogX. A core concept for the spatial model is that of *places* which serve as the smallest unit of space which the majority of the rest of the system should be reasoning about. Closely related to this is the work on recognising and categorising such places. We also present our work on defining quantitative measures for topological spatial relations, so far fully developed for the relation "on". These spatial relations will be used, for example, when abstracting spatial knowledge and when verbalising such knowledge. We also show how these relations can be used to implement so called indirect object search in a principled way. Finally we include a number of theoretical results regarding the advantage of using a robocentric representation of space – one of the key ideas in our spatial model, where a robocentric map would contain a detailed description of the robot's surroundings, outside which the robot would start to forget/abstract away information.

---

## Executive Summary

This report is the first of the deliverables in CogX from WP3 which deals with representations for qualitative spatial cognition. The first year was spent on requirements analysis and designing the interface for the spatial representation for our cognitive system. A mock-up system, presenting the rest of the system with a working interface similar to the real one, was used in order to run the overall system during the first year. During the second year we have been working on some of the components that will constitute the final system. This report covers the progress to the end of the second year's reporting period. More progress in terms of implementation is expected until the review meeting. This report does not attempt to describe the implementation part of the work package so much as the scientific issues and it covers the material that has already been published or submitted to date.

Relating the work over the first two years to the workplan, the main focus has been on the overall spatial representation (Task 3.1) and how to reference spatial entities (Task 3.2). Some initial work has also started around the issues of how and what to represent in long-term and short-term spatial memory (Task 3.3). The report focuses on the work on Tasks 3.1 and 3.2 as the work on 3.3 is still to be considered preliminary.

One of the fundamental research questions that we want to investigate with our spatial design is to what extent the system can operate without a representation that maintains all its knowledge in a global metric frame of reference. Our hypothesis is that this is not necessary and we argue that in fact not using such a global frame makes the problem more tractable and the system more robust. This said we still believe that metric information plays an important role on a local scale. Given that most of the existing techniques for components assume a global metric reference frame it has resulted in a bit more implementation work than initially anticipated.

In addition to the overall design of the spatial representation and the issues of integration with the rest of the system such as the architecture (WP1), vision and perception (WP2), planning and execution (WP4) and dialogue (WP6), we have been working mainly along four tracks: i) methods for recognising and categorising spatial regions with categories such as kitchen, corridor and office, ii) how to make use of spatial relations when describing, reasoning, communicating and storing spatial knowledge, iii) how to efficiently find objects using both top-down and bottom-up information in an efficient way and iv) metric level navigation.

Finally, the work has resulted in a number of high quality publications, among them 4 journal articles. The content of these and how they relate to CogX as a whole and this work-package in particular are outlined at the end of this report.

## Role of spatial cognition in CogX

The overarching goal in CogX is for the robot to self-understand and self-extend. An important part of this is the robot's understanding of the space around it. It needs to be able to navigate, that is know where it is and how to get from one point to another and it needs to know how to perform useful tasks in said environment. We are dealing with a cognitive agent that is assumed to be sharing its environment with people and therefore the ability to interact with people is very important. The latter brings with it requirements on the spatial representation and understanding that go beyond simple navigation tasks. The robot needs to be able to exchange knowledge that is related to the environment with people and other agents, it needs to be able to plan tasks, reason about high level, human-type concepts such as objects, rooms, etc. WP3 is about designing, building and maintaining a spatial representation that supports these requirements.

One of the main goals in CogX is to endow the robot with the ability to identify gaps in its knowledge and thereby gain a form of self-understanding. When it comes to self-understanding in the context of WP3, we have identified a number of gaps at different levels of abstraction in the spatial representation which are described in Annexes 2.1.1/[42], 2.1.2/[43], 2.1.3/[56]. In the publications that form the foundation for this deliverable we deal with a number of these gaps in more detail and we discuss how to fill them and thus self-extend. In Annexes 2.2.2/[40] and 2.2.3/[41] we deal with the knowledge gaps regarding room categories which play an important role in our system where interaction with humans is central. In Annexes 2.4.1/[1] and 2.4.2/[2] we describe how the robot can fill a gap in knowledge about the location of object that may arise e.g. when a human orders the robot to "find object X". In this process we explictly reason about our knowledge and plan the best next sensing action to find the object. In Annex 2.5.1/[6] we perform the same type of knowledge-gathering but from a more theoretical standpoint. Annexes 2.5.2/[7] and 2.5.3/[8] address the problem of gathering knowledge about the geometry of large scale space (the localization and mapping problem in robotics). Annex 2.5.4/[5] shows how finite set statistics can be used to localize a robot without prior information about the location of the robot which is vital for most tasks that the robot takes on.

## Contribution to the CogX scenarios and prototypes

The work in WP3 is currently mostly related to the Dora scenario (one of three scenarios/demonstrators in WP7: Scenario-based integration) since the focus in WP3 is on large-scale space which fits well with Dora. However, the aim is to work towards a seamless integration with the small-scale

representation of space that is used in the George scenario. The work on navigation in WP3 can be used in the George scenario during the third year if that scenario requires the robot to move to acquire novel views of objects that are observed or hypothesised about.

In the Dora scenario, the work in WP3 is the "low-level" corner stone in that it provides the robot with the ability to move about in the environment, and build up a representation that the rest of the system can make use of when, for example, planning actions and reasoning about space. There is a very tight connection with the work in WP6 on adaptive situated dialogue processing as most of the communication with the human, in the Dora scenario, is about space in one way or another. Questions that cut across these two work packages and have contributed requirements to the design are, for example, how to represent the spatial information to facilitate efficient verbalisation and how to represent spatial knowledge that is provided by the human.

# 1 Tasks, objectives, results

## 1.1 Planned work

Spatial knowledge constitutes a fundamental component of the knowledge base of a mobile agent, such as Dora, and many functionalities directly depend on the structure of the spatial knowledge representation, ranging from navigation, to spatial understanding and communication. These include localisation, mapping, way-finding and autonomous exploration, but also understanding and exploiting semantics associated with space, human-like conceptualisation and categorisation of and reasoning about spatial units and their relations, human-robot communication, action planning, object finding and visual servoing, and finally storing and recalling episodic memories.

WP3 is about qualitative spatial cognition, by which we mean that, in addition to quantitative aspects, the system must also capture qualitative aspects. We will work with different scenarios, ranging from the robot being given only some basic object recognition skills and being asked to build up the spatial knowledge from scratch, to situations where a substantial amount of innate knowledge is given *a priori* to be able to study some aspects that would otherwise require the system to have operated for a very long time. We will also make use, when suitable, of knowledge gathered from various databases which help provide the system with "common sense" knowledge.

The tasks from the workplan we planned to work on during the first two years were:

**Task 3.1: Spatial modelling.** *The goal is to develop a framework that allows for a hybrid representation where objects and traditional metric spatial models coexist.*

**Task 3.2: Spatial referencing.** *The goal is to investigate what objects and other entities in the map should be referenced and how.*

**Task 3.3: Short-term vs long-term spatial memory.** *The goal is to investigate how spatial knowledge should be represented to support both short-term and long-term storage and access.*

Task 3.1 was intended to lay the foundation for the work in WP3 by defining a spatial representation that caters to the requirements given by the desired functionality of the robot and the rest of the components in the system. The task was planned to end by the second year but it is likely that the design will be updated as the project progresses and new requirements surface. It is very hard to find all of the requirements before they have appeared in actuality, especially from components that are not directly related to the spatial representation. For example, as the scenarios

George and Dora come closer to each other it is likely that some additional requirements will appear.

Task 3.2 is part of our strategy to handle abstraction of spatial information and facilitate natural communication between humans and robotic systems. We believe that it is crucial for a cognitive system to be able to both analyse a scene to determine the spatial relations between entities and to be able to generate a prior for how a certain described scene might appear when perceived. Task 3.3 is closely related to Task 3.2 in that we hope that the results from Task 3.2 can help structure part of the long-term memory.

In the next section we describe how we achieved the goals for the first two years.

## 1.2 Actual work performed

Below we summarise the achievements for the individual tasks. We provide a somewhat disproportionately long description of the design of the spatial structure motivated by the fact that it is underpinning the entire future system design.

### 1.2.1 Task 3.1: Spatial Modelling

As with all designs we started by looking at the requirements that are placed on our spatial model. Following Davis' [17] analysis the model must provide: *a)* A model of the real world, to allow reasoning also beyond the field of view of the sensors. As pointed out by many such a model is always going to be imperfect and due to dynamics and the complexity of the real world also eventually invalid. *b)* A definition of the aspects of the world that should be represented and at what level. *c)* Definitions for the reasoning that can be performed within the framework and the possible inferences and their outcomes. *d)* A structure that makes processing of the information computationally tractable with limited resources. *e)* A medium of communication between the agent and a human. *f)* A medium for communication between components in the system.

With these functional requirements in mind we have designed a representation presented in Annexes 2.1.1/[42], 2.1.2/[43], 2.1.3/[56] as well as in [44] and briefly in [55]. It is designed for representing complex, cross-modal, spatial knowledge that is inherently uncertain and dynamic. Our main hypothesis is that it is not a good idea to aim to represent the world as accurately as possible, due to its complexity and dynamic nature. We instead argue that the representation should be coarse and only as accurate as needed. Uncertainties and gaps in the spatial knowledge must also be represented explicitly. An example of such an explicit representation of gaps is so called *placeholders* for representing unexplored space. We represent categorical knowledge separately from location specific information.

| Property | Sensory Layer | Place Layer | Categorical Layer | Conceptual Layer |
|---|---|---|---|---|
| Aspects represented | Accurate geometry and appearance | Local spatial relations, coarse appearance & geometry | Perceptual categorical knowledge | High-level spatial concepts / Links concepts ↔ entities |
| Agent's position | Pose within the local map | Place ID | Relationship to the categorical models | Expressed in terms of high level spatial concepts |
| Spatial scope | Small-scale, local | Large-scale | Global | Global |
| Knowledge persistence | Short-term | Long-term | Very long-term | Life-long / Very long-term |
| Knowledge decay | Replacement | Generalization, forgetting | Generalization | None / Forgetting |
| Information flow | Bottom-up | Primarily bottom-up | Primarily bottom-up | Top-down and bottom-up |

Table 1: Comparison of properties of the four layers of the spatial representation.

Examples of categorical models are the appearance of places [39] and of objects [37]. Also, as the representation is a key part in human-robot interaction, we model correspondence between the represented symbols and human concepts of space. This information is used for example to generate and resolve spatial referring expressions [58]. This is an example of the close coupling between WP3 and WP6. Results from this year can be found in Annex 2.2.1 in DR.6.2 [57].

Some of the work we have done is directly related to modelling, while other parts use the spatial representation and build up the information in it. In Annexes 2.4.1/[1] and 2.4.2/[2] we describe how the robot can fill the gap in knowledge about the location of object that can arise for example when a human orders the robot to "find object X". In Annex 2.5.1/[6] a theoretical treatment of the same problem is presented.

Our representation can be divided into four layers which capture different aspects of the world. Figure 1 provides an illustration of the the structure and Table 1 summarises the properties characterising each layer. Below, a brief description of each layer will be given.

**Sensory Layer** The sensory layer maintains a detailed model of the environment in close proximity to the robot. Information beyond a certain distance is forgotten and replaced by new information, making the sensory

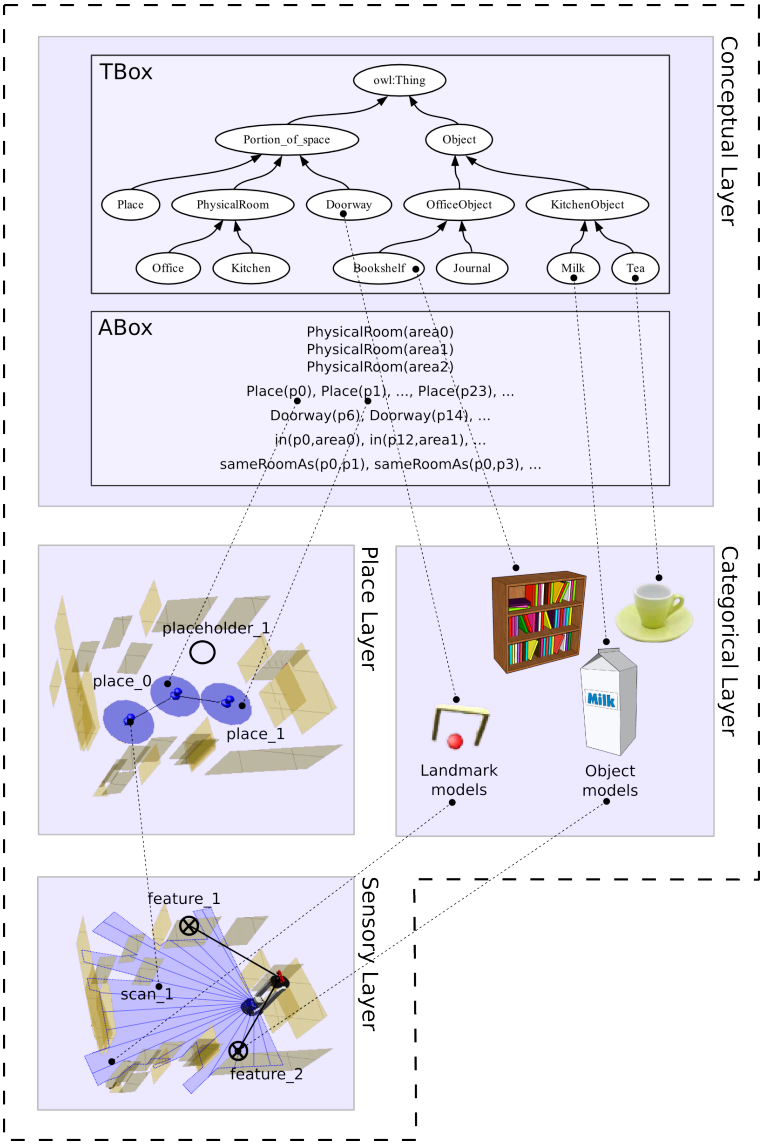



Figure 1: The layered structure of the spatial representation. The position of each layer within the representation corresponds to the level of abstraction of the spatial knowledge.

layer operate as a sliding window. The sensory layer stores low-level features together with their exact position including uncertainty. The sensory layer also provides the low level robotic movement systems with data for deriving basic control laws, e.g., for obstacle avoidance or visual servoing.

Annexes 2.5.2/[7] and 2.5.3/[8] describes the benefits of a robocentric representation. Annex 2.5.4/[5] explains how to fill gaps in robot pose information by the use of finite set statistics.

**Place Layer** The place layer represents the world as a collection of basic spatial entities called *places* as well as their spatial relations. The places are intended to be the smallest units of space that the rest of the system reasons about. The place layer explicitly represents *gaps in knowledge about explored space* by instantiating so called placeholders.

Annexes 2.2.1/[38], 2.2.2/[40] and 2.2.3/[41] all deal with place recognition and classification.

**Categorical Layer** This layer contains the categorical models for the robot's sensory information. The information is intended to be general across all environments that the robot has come across. This is the place where models of objects are defined in terms of low-level features. In the case of models that correspond to human concepts, they can be learnt in a supervised fashion, using a top-down supervision signal or from databases.

The papers that were mentioned under the place layer are also connected with the categorical layer in that they discuss how to train categorical models for places.

**Conceptual Layer** The conceptual layer acts as the bridge between the spatial representation of the robot and that of humans, and makes use of human-compatible concepts. Information at this level is symbolic and contains links to the lower levels of the spatial models. The conceptual layer represents a segmentation of the environment into rooms and can supply default assumptions about which kinds of objects are likely to be found in which kinds of rooms.

### 1.2.2 Task 3.2: Spatial Relations

A human might describe a certain office as having a desk *in* the room, *near* the window and having a computer monitor *on* the desk and a chair *at* the desk. A robot on the other hand would typically have the same scene represented in terms of metric coordinates. If spatial knowledge is to be passed between the robot and the human through spoken words or text the robot must be able to verbalise its knowledge in a way that is understandable to the human and, conversely, understand what the human says and transform that into a possibly different internal representation.

Having this ability would also provide qualitative abstractions that facilitate learning and reasoning and guide top-down processes such as e.g. visual object search.

During the second year, in the context of task 3.2, we have been examining functional spatial relations starting with that of mechanical support, which in English corresponds to the preposition "on". We have contributed a novel and general perceptual measure that allows a robot to analyse a scene in terms of this relation in practice and also show how this can be used to generate a prediction for the distribution of objects in a scene given (uncertain) knowledge about their spatial relations (see Annex 2.3.1/[48]).

We have also shown how such a framework can be used to guide visual object search (Annex 2.4.2/[2]). This allowed for a principled way of making use of the concept of indirect search [21]. In indirect search the idea is that in order to find for example a stapler it is often easier to look for the desk first. This is true for humans and, we believe, even more so for a robot with much more limited perceptual abilities where every chance to cut down the search space has to be seized. The spatial relations provide the means to generate a prior over the locations of objects given common sense knowledge such as "staplers are typically found on tables".

### 1.2.3 Task 3.3: Short-term vs long-term

Although the computer's ability to store digital information is improved every year it is still not possible for the robot to store all the information that it gathers during its life-time. In this task we want to investigate how to represent information that is meant for short-term and long-term storage and what types of information go where. The work on this task has so far been carried out in parallel with task 3.2. We believe that we can make use of spatial relations to describe and represent long-term common sense type spatial knowledge. We will have the robot try to learn such spatial knowledge over long periods of time and also investigate ways to extract such spatial knowledge from databases. As already mentioned, the spatial relations provide a means for more efficient learning of certain aspects of space. In many cases it will be enough for the robot to learn that the object is typically found on the table rather than trying to learn the full distribution over space. In the latter case one would need extremely large amounts of data.

There is an intimate connection between the work on the spatial model itself and Task 3.3. Table 1 shows how the persistence of the information varies with the layer in the spatial representation. Using our design philosophy, one should represent things at the highest level of abstraction and the higher the abstraction the longer knowledge can be kept since the more likely it is to be valid over long periods of time.

As was said above when describing the categorical layer in the spatial

model, it contains persistent long-term models. The acquisition of such models is therefore intimately linked with Task 3.3. The work presented in Annex 2.2.3/[41] deals with the issue of incremental learning of such categorical models for places. One of the contributions of the paper is showing how information has to be forgotten also in the long-term memory to adapt to the slow changes in the environment that might occur over time scales of months.

## 1.3 Relation to state-of-the-art

Below we briefly discuss how the obtained results relate to the current state-of-the-art. We refer the reader to the annexes for more in-depth discussions.

### 1.3.1 Task 3.1: Spatial modeling

All but a few of the mobile robots brought into being to date have been relying on two-dimensional geometric and global maps. The sensor of choice was first sonars [34, 12, 54] and later lasers [51, 10, 22]. Lately, more and more of the work relies on cameras as the main sensory modality [18, 33, 47, 19, 23, 11, 13]. So far the majority of the work has followed the same direction as with the laser scanners and sonar sensors, that is, trying to create accurate metric models of the environment. There are also examples where the information about appearance given by visual data has been used to build topological maps. Places with a distinct appearance become nodes in a graph. In fact, some of the really early examples of robot mapping [9] are based on visual data. Additional work includes for example [26, 52, 16, 15]. All of the work mentioned above treats the problem of mapping as a problem of representing space to perform navigation tasks. For the applications we want to support, this is not enough. If we want the robot to perform human-like tasks and assist humans in complex and dynamic environments the spatial representations need to support this. We need to expand the scope of the spatial model so that it can carry out the role of knowledge base for spatial reasoning, understanding, interaction, etc. In addition, it is not clear that the level of detail offered by most global metric maps is necessary, or even desirable. Although the level of detail in the most topological maps might be right they are still geared towards navigation and do not support all desired tasks.

The idea of the *cognitive map* [24] has been used to inspire work in spatial representation before [28, 25]. The cognitive map contains the spatial knowledge that the agent (human or robot) has gathered about the world. Key references in this area in robotic is the work by Kuipers *et al.* [32, 29, 30, 31] on the TOUR model and the so called Spatial Semantic Hierarchy (SSH) which can be seen as an implementation of these ideas. An approach akin to the SSH is the *Route Graph* model [27]. According to [36], humans have a

more qualitative, topological perspective on spatial organisation. The focus in the work by Kuipers is procedural route descriptions and on understanding how much the agent can learn and do without providing it with any or little prior knowledge. To facilitate smooth and natural interaction between humans and robots the representation of space should help bridge the gap between how humans and robots perceive space. The work mentioned above are examples in this direction. Over the last years a number of systems have been presented where the robot can acquire and facilitate semantic information [50, 49, 20, 53]. The work presented in [50, 49] is mostly concentrated on linguistic interaction with a human and the robot is not using its sensors to retrieve semantic information. The anchoring approach, presented in [20], deals mostly with the problem of integrating semantic and spatial levels using anchoring. Vasudevan *et al.* [53] suggest a hierarchical probabilistic representation of space based on objects. In [45] a model for representing places based on constellations of objects is presented. The *Hybrid Spatial Semantic Hierarchy* (HSSH), introduced by Beeson *et al.* [4], allows a mobile robot to describe the world using different representations, each with its own ontology.

### 1.3.2 Tasks 3.2-3

Our work in Task 3.2 is by no means the first attempt to quantify spatial relations. In [46] the *Attention Vector Sum* is proposed as a practical numerical measure of how acceptable a particular spatial relation is for describing a scene (only 2D), and this model is compared to actual human responses. Lockwood et al [35] present a system which learns to distinguish between "in", "on", "above", "below" and "left" from sketched images of basic figures. However, this too is restricted to a 2-dimensional world. Topological relations specifically are surveyed in [14]. *Region connection calculus* and its variants provide a language for expressing qualitative relationships between regions, such as containment, tangential contact etc. Relations are of an all-or nothing nature; and they represent objective, geometrical as opposed to perceptual or functional attributes. The work mentioned above is not well suited as is for practical robotics application because of its emphasis on pure geometry, typically in 2 dimensions as well. Our work, in contrast, takes a novel, functional approach. For example, in the case of the "on" relation we base it on a single fundamental, objective mechanical property, that of support. Another contribution of our work lies in treating all the objects as entire bodies rather than simplifying them into points, a simplification which ignores important aspects of the relations.

# 2 Annexes

The annexes consist almost exclusively of peer-reviewed papers (4 journal papers, 9 conference papers and 1 technical report). They have been divided according to the task that they are most closely related to. Many papers cut across several tasks; for example, the work on incremental learning of categorical models of places is both core to our spatial model and also highly related to the issue of adapting the long-term memory and forgetting information that is no longer relevant to describe the environment. Some of the papers describe the spatial representation and the models therein, while others describe processes for building the representations and populating them with information.

**Task 3.1**

- Annex 2.1.1 Pronobis et al IAS 2010 (accepted) [43]
- Annex 2.1.2 Pronobis et al ICAR 2009 [42]
- Annex 2.1.3 Zender 2010 (technical report) [56]
- Annex 2.2.1 Pronobis et al RAS 2010 [38]
- Annex 2.2.2 Pronobis et al IJRR 2010 [40]
- Annex 2.4.1 Aydemir et al ICRA 2010 [1]
- Annex 2.5.1 Bishop et al ECC 2009 [6]
- Annex 2.5.2 Bishop et al ICRA 2009 [7]
- Annex 2.5.3 Boberg et al ISSNIP 2009 [8]
- Annex 2.5.4 Bishop et al FUSION 2010 (accepted) [5]
- Annex 2.5.5 Basiri et al Systems & Control Letters 2010 [3]

**Task 3.2**

- Annex 2.3.1 Sjöö et al IROS 2010 (accepted) [48]
- Annex 2.4.2 Aydemir et al IAS 2010 (accepted) [2]

**Task 3.3**

- Annex 2.2.3 Pronobis et al IMAVIS 2010 [41]

An alternative way to divide the publications is to look at the topic of the paper. This is how we choose to present the annexes as we believe that it makes it easier to read them.

- Structure of the Spatial Representation
- Multi-modal Place Categorization
- Spatial Relations
- Object search
- Metric navigation

## 2.1 Structure of the Spatial Representation

The papers and reports in this section describe the overall design of the spatial representation. Papers [42, 43] are both accepted at conferences. The paper in annex 2.1.3 is a technical report.

### 2.1.1 Pronobis et al, "Representing Spatial Knowledge in Mobile Cognitive Systems", (IAS 2010)

**Bibliography** A. Pronobis, K. Sjöö, A. Aydemir, A. Bishop and P. Jensfelt, "Representing Spatial Knowledge in Mobile Cognitive Systems", 11th International Conference on Intelligent Autonomous Systems (IAS-11), August 2010, Ottawa, Canada

**Abstract** A cornerstone for cognitive mobile agents is to represent the vast body of knowledge about space in which they operate. In order to be robust and efficient, such representation must address requirements imposed on the integrated system as a whole, but also resulting from properties of its components. In this paper, we carefully analyze the problem and design a structure of a spatial knowledge representation for a cognitive mobile system. Our representation is layered and represents knowledge at different levels of abstraction. It deals with complex, cross-modal, spatial knowledge that is inherently uncertain and dynamic. Furthermore, it incorporates discrete symbols that facilitate communication with the user and components of a cognitive system. We present the structure of the representation and propose concrete instantiations.

**Relation to WP** This is the first of two publications that describe the overall spatial design (Taks 3.1) and is thus at the very heart of WP3. This paper describes the analysis of the problem and the requirements.

### 2.1.2 Pronobis et al, "A Framework for Robust Cognitive Spatial Mapping", (ICAR 2009)

**Bibliography** A. Pronobis, K. Sjöö, A. Aydemir, A.N. Bishop and Patric Jensfelt, "A Framework for Robust Cognitive Spatial Mapping", In Proc. of the International Conference on Advanced Robotics (ICAR'09), June 2009, Munich, Germany

**Abstract** Spatial knowledge constitutes a fundamental component of the knowledge base of a cognitive, mobile agent. This paper introduces a rigorously defined framework for building a cognitive spatial map that permits high level reasoning about space along with robust navigation and localization. Our framework builds on the concepts of places and scenes expressed in terms of arbitrary, possibly complex features as well as local spatial relations. The resulting map is topological and discrete, robocentric and specific to the agent's perception. We analyze spatial mapping design mechanics in order to obtain rules for how to define the map components and attempt to prove that if certain design rules are obeyed then certain map properties are guaranteed to be realized. The idea of this paper is to take a step back from existing algorithms and literature and see how a rigorous formal treatment can lead the way towards a powerful spatial representation for localization and navigation. We illustrate the power of our analysis and motivate our cognitive mapping characteristics with some illustrative examples.

**Relation to WP** This is the second of two publications that are at the core of WP3 in that they describe the overall design of our spatial representation (Task 3.1). This paper focuses on the place layer.

### 2.1.3 Zender, "Multi-Layered Conceptual Spatial Mapping", (technical report)

**Bibliography** H. Zender, "Multi-Layered Conceptual Spatial Mapping", technical report, DFKI GmbH, 2010

**Abstract** In this paper, we identify structuring of space and categorization of large-scale space as two important aspects of spatial understanding for embodied cognitive systems. In order to enable an autonomous agent to engage in a situated dialogue about its environment, it needs to have a human-compatible spatial understanding, whereas autonomous behavior, such as navigation, requires the agent to have access to low-level spatial representations. Addressing these two challenges, we present an approach to multi-layered conceptual spatial mapping. We embed our work in a discussion of relevant research in human spatial cognition and mobile robot mapping.

**Relation to WP** This technical report also gives a description of the structure of the spatial model (Task 3.1). It relates the research on spatial mapping undertaken in WP3 with the work on spatially situated dialogue processing in WP6.

## 2.2 Multi-modal Place Categorization

One of the core concepts in our spatial model is that of places. The publications in this section present work on classification of places, incremental learning of such models and a benchmark for visual place recognition.

### 2.2.1 Pronobis et al, "A Realistic Benchmark for Visual Indoor Place Recognition", (RAS Jan 2010)

**Bibliography** A. Pronobis, B. Caputo, P. Jensfelt and H. I. Christensen, "A realistic benchmark for visual indoor place recognition", Robotics and Autonomous Systems, Jan 2010, 58:1, pp. 81–96

**Abstract** An important competence for a mobile robot system is the ability to localize and perform context interpretation. This is required to perform basic navigation and to facilitate local specific services. Recent advances in vision have made this modality a viable alternative to the traditional range sensors, and visual place recognition algorithms emerged as a useful and widely applied tool for obtaining information about robot's position. Several place recognition methods have been proposed using vision alone or combined with sonar and/or laser. This research calls for standard benchmark datasets for development, evaluation and comparison of solutions. To this end, this paper presents two carefully designed and annotated image databases augmented with an experimental procedure and extensive baseline evaluation. The databases were gathered in an uncontrolled indoor office environment using two mobile robots and a standard camera. The acquisition spanned across a time range of several months and different illumination and weather conditions. Thus, the databases are very well suited for evaluating the robustness of algorithms with respect to a broad range of variations, often occurring in real-world settings. We thoroughly assessed the databases with a purely appearance-based place recognition method based on support vector machines and two types of rich visual features (global and local).

**Relation to WP** This paper presents two datasets that are extensively used when implementing and testing the algorithms in the instantiations of the spatial model (Task 3.1). It is mostly used in the work on the place layer but the work on the categorical and sensory layer also benefit greatly from the data.

### 2.2.2 Pronobis et al, "Multi-modal Semantic Place Classification", (IJRR Feb 2010)

**Bibliography** A. Pronobis, O. Martinez Mozos, B. Caputo and P. Jensfelt, "Multi-modal semantic place classification", The International Journal of Robotics Research (IJRR), Feb 2010, 29:2-3, pp. 298–320

**Abstract** The ability to represent knowledge about space and its position therein is crucial for a mobile robot. To this end, topological and semantic descriptions are gaining popularity for augmenting purely metric space representations. In this paper we present a multi-modal place classification system that allows a mobile robot to identify places and recognize semantic categories in an indoor environment. The system effectively utilizes information from different robotic sensors by fusing multiple visual cues and laser range data. This is achieved using a high-level cue integration scheme based on a Support Vector Machine (SVM) that learns how to optimally combine and weight each cue. Our multi-modal place classification approach can be used to obtain a real-time semantic space labeling system which integrates information over time and space. We perform an extensive experimental evaluation of the method for two different platforms and environments, on a realistic off-line database and in a live experiment on an autonomous robot. The results clearly demonstrate the effectiveness of our cue integration scheme and its value for robust place classification under varying conditions.

**Relation to WP** As already described places are a core concept in the spatial model. This paper presents a method for classifying places based on multi-modal input (laser and vision). The learning of the models in the work is supervised by a human. Currently we are extending this work to be able to do automatic segmentation of space into places as well. This work is most closely related to Task 3.1.

### 2.2.3 Pronobis et al , "The More You Learn, the Less You Store: Memory-controlled Incremental SVM for Visual Place Recognition", (IMAVIS Mar 2010)

**Bibliography** A. Pronobis, J. Luo and B. Caputo, "The More You Learn, the Less You Store: Memory-controlled Incremental SVM for Visual Place Recognition", Image and Vision Computing (IMAVIS), March 2010

**Abstract** The capability to learn from experience is a key property for autonomous cognitive systems working in realistic settings. To this end, this paper presents an SVM-based algorithm, capable of learning model representations incrementally while keeping under control memory requirements. We combine an incremental extension of SVMs [43] with a method reducing the number of support vectors needed to build the decision function without any loss in performance [15] introducing a parameter which permits a user-set trade-off between performance and memory. The resulting algorithm is able to achieve the same recognition results as the original incremental method while reducing the memory growth. Our method is especially suited to work for autonomous systems in realistic settings. We present experiments on two common scenarios in this domain: adaptation in presence of dynamic changes and transfer of knowledge between two different autonomous agents, focusing in both cases on the problem of visual place recognition applied to mobile robot topological localization. Experiments in both scenarios clearly show the power of our approach.

**Relation to WP** One of the long term goals of the project, this work-package and most of robotics in general is life-long learning. This paper presents work on how the robot can adapt its place models over time as new data is made available. This could be one of the mechanisms that will be used in the future system to maintain the place models. This work has close ties to Task 3.1 but even more so with Task 3.3 in that it describes a way to deal adapt long-term spatial knowledge.

## 2.3 Spatial Relations

This sections represents the published work on spatial relations. We are currently working on extending it to other topological spatial relations besides "on".

### 2.3.1 Sjöö et al, "Mechanical support as a spatial abstraction for mobile robots", (IROS 2010)

**Bibliography** K. Sjöö and A. Aydemir and P. Jensfelt, "Mechanical support as a spatial abstraction for mobile robots", Proc. of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'10), Oct 2010

**Abstract** Motivated by functional interpretations of spatial language terms, and the need for cognitively plausible and practical abstractions for mobile service robots, we present a spatial representation based on the physical support of one object by another inspired by the preposition "on". A perceptual model for evaluating this relation is suggested, and experiments simulated as well as using a real robot are presented. We indicate how this model can be used for important tasks such as communication of spatial knowledge, abstract reasoning and learning, exemplifying this in the context of direct and indirect visual search. We also demonstrate the model experimentally, showing that it produces intuitively feasible results from visual scene analysis as well as synthetic distributions that can be put to a number of uses.

**Relation to WP** A quantitative functional measure of the spatial relation "on" is introduced in the paper. This work is directly related to Task 3.2 on spatial relations but also to Task 3.3 by treating the issue of abstracting information for long-term storage.

## 2.4 Active Object Search

We belive objects play an important role in spatial cognition. The papers in this section represents the result of our published work on active object search.

### 2.4.1 Aydemir et al, "Simultaneous Object Class and Pose Estimation for Mobile Robot ic Applications with Minimalistic Recognition", (ICRA 2010)

**Bibliography** A. Aydemir, A.N. Bishop and P. Jensfelt, "Simultaneous Object Class and Pose Estimation for Mobile Robotic Applications with Minimalistic Recognition", Proc. of the IEEE International Conference on Robotics and Automation (ICRA'10), May 2010, Anchorage, Alaska, USA

**Abstract** In this paper we address the problem of simultaneous object class and pose estimation using nothing more than object class label measurements from a generic object classifier. We detail a method for designing a likelihood function over the robot configuration space. This function provides a likelihood measure of an object being of a certain class given that the robot (from some position) sees and recognizes an object as being of some (possibly different) class. Using this likelihood function in a recursive Bayesian framework allows us to achieve a kind of spatial averaging and determine the object pose (up to certain ambiguities to be made precise). We show how inter-class confusion from certain robot viewpoints can actually increase the ability to determine the object pose. Our approach is motivated by the idea of minimalistic sensing since we use only class label measurements albeit we attempt to estimate the object pose in addition to the class.

**Relation to WP** We need to find the objects before we can use them in our spatial model. This paper addresses the problem of finding objects with the use of a recognition system that only provides yes/no output to the question "is the object in the image". The paper is to be considered more of a theoretical study on what can be achieved with very limited perceptual information rather than a suggestion for how to actually implement an efficient search system. A real system should in general make use of as much of the available information as possible and only trade off using information if it comes at an extra cost. This work is most closely related to Task 3.1.

### 2.4.2 Aydemir et al, "Object search on a mobile robot using relational spatial information", (IAS 2010)

**Bibliography** A. Aydemir, K. Sjöö and P. Jensfelt, "Object search on a mobile robot using relational spatial information", Proc. of the 11th Int Conference on Intelligent Autonomous Systems (IAS-11), August 2010, Ottawa, Canada

**Abstract** We present a method for utilising knowledge of qualitative spatial rela- tions between objects in order to facilitate efficient visual search for those objects. A computational model for the relation is used to sample a probability distribution that guides the selection of camera views. Specifically we examine the spatial relation on, in the sense of physical support, and show its usefulness in search experiments on a real robot. We also experimentally compare different search strategies and verify the efficiency of so-called indirect search.

**Relation to WP** This paper ties together the work on active visual search from WP2 with the work on spatial relations from Task 3.2 in WP3.

## 2.5 Metric Navigation

### 2.5.1 Bishop & Jensfelt, "Stochastically Convergent Localization of Objects by Mobile Sensors and Actively Controllable Relative Sensor-Object Pose", (ECC 2009)

**Bibliography** A.N. Bishop and P. Jensfelt, "Stochastically Convergent Localization of Objects by Mobile Sensors and Actively Controllable Relative Sensor-Object Pose", Proc. of European Control Conference (ECC'09), 2009, Budapest, Hungary

**Abstract** The problem of object (network) localization using a mobile sensor is examined in this paper. Specifically, we consider a set of stationary objects located in the plane and a single mobile nonholonomic sensor tasked at estimating their relative position from range and bearing measurements. We derive a coordinate transform and a relative sensor-object motion model that leads to a novel problem formulation where the measurements are linear in the object positions. We then apply an extended Kalman filter-like algorithm to the estimation problem. Using stochastic calculus we provide an analysis of the convergence properties of the filter. We then illustrate that it is possible to steer the mobile sensor to achieve a relative sensorobject pose using a continuous control law. This last fact is significant since we circumvent Brockett's theorem and control the relative sensor-source pose using a simple controller.

**Relation to WP** In the sensory layer one of the main activities is to estimate the poses of objects in the robot's surroundings. This paper presents a theoretical study into the problem of how to move the sensor to achieve a certain sensor-object pose. The work is associated with Task 3.1.

### 2.5.2 Bishop & Jensfelt, "A Stochastically Stable Solution to the Problem of Robocentric Mapping", (ICRA 2010)

**Bibliography** A.N. Bishop and P. Jensfelt, "A Stochastically Stable Solution to the Problem of Robocentric Mapping", Proc. of the Interntional Conference on Robotics and Automation (ICRA'09), 2009, Kobe, Japan

**Abstract** This paper provides a novel solution for robocentric mapping using an autonomous mobile robot. The robot dynamic model is the standard unicycle model and the robot is assumed to measure both the range and relative bearing to the landmarks. The algorithm introduced in this paper relies on a coordinate transformation and an extended Kalman filter like algorithm. The coordinate transformation considered in this paper has not been previously considered for robocentric mapping applications. Moreover, we provide a rigorous stochastic stability analysis of the filter employed and we examine the conditions under which the mean-square estimation error converges to a steady-state value.

**Relation to WP** One of the central ideas in the sensory layer of the spatial model is that the detailed metric information should be maintained in a robocentric representation. This paper presents the theoretical justification for this in terms of convergence and stability of the solution and is part of the work in Task 3.1.

### 2.5.3 Boberg et al, "Robocentric Mapping and Localization in Modified Spherical Coordinates with Bearing Measurements", (ISSNIP 2009)

**Bibliography** A. Boberg and A.N. Bishop and P. Jensfelt, "Robocentric Mapping and Localization in Modified Spherical Coordinates with Bearing Measurements", Proc. of the Fifth International Conference on Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP 2009), Decemeber 2009, Melbourne, Australia

**Abstract** In this paper, a new approach to robotic mapping is presented that uses modified spherical coordinates in a robotcentered reference frame and a bearing-only measurement model. The algorithm provided in this paper permits robust delayfree state initialization and is computationally more efficient than the current standard in bearing-only (delay-free initialized) simultaneous localization and mapping (SLAM). Importantly, we provide a detailed nonlinear observability analysis which shows the system is generally observable. We also analyze the error convergence of the filter using stochastic stability analysis. We provide an explicit bound on the asymptotic mean state estimation error. A comparison of the performance of this filter is also made against a standard world-centric SLAM algorithm in a simulated environment.

**Relation to WP** This paper presents one example instantiation of a robocentric mapping system using vision-only information. The paper provides further theoretical justifications to the idea of using a robocentric representation (Task 3.1).

### 2.5.4 Bishop & Jensfelt, "Global robot localization with random finite set statistics", (FUSION 2010)

**Bibliography** A.N. Bishop and P. Jensfelt, "Global robot localization with random finite set statistics", Proc. of 13th International Conference on Information Fusion (FUSION), July 2010, Edinburgh, UK

**Abstract** We re-examine the problem of global localization of a robot using a rigorous Bayesian framework based on the idea of random finite sets. Random sets allow us to naturally develop a complete model of the underlying problem accounting for the statistics of missed detections and of spurious/erroneously detected (potentially unmodeled) features along with the statistical models of robot hypothesis disappearance and appearance. In addition, no explicit data association is required which alleviates one of the more difficult sub-problems. Following the derivation of the Bayesian solution, we outline its first-order statistical moment approximation, the so called probability hypothesis density filter.n c We present a statistical estimation algorithm for the number of potential robot hypotheses consistent with the accumulated evidence and we show how such an estimate can be used to aid in re-localization of kidnapped robots. We discuss the advantages of the random set approach and examine a number of illustrative simulations.

**Relation to WP** This paper presents work on localization which is one of the fundamental competencies in a robotics system. What makes the work interesting for CogX is that it provides a framework for incorporating linguistic information into the localization system in a principled way. This means, for example, that the robot could make use of statements such as "Your are in front of the door" and thus connect nicely with the work in, for example, WP6. The work is part of Task 3.1.

### 2.5.5 Basiri et al, "Distributed Control of Triangular Formations with Angle-Only Constraints", (Systems & Control Letters 2010)

**Bibliography** M. Basiri, A.N. Bishop and P. Jensfelt, "Distributed Control of Triangular Formations with Angle-Only Constraints", Systems & Control Letters, Feb 2010, issue 2, pp.147–154

This article was listed at 4th place on a list with the Top 25 Hottest Articles[1] in the Systems & Control Journal.

**Abstract** This paper considers the coupled formation control of three mobile agents moving in the plane. Each agent has only local inter-agent bearing knowledge and is required to maintain a specified angular separation relative to both neighbor agents. Assuming the desired angular separation of each agent relative to the group is feasible, then a triangle is generated. The control law is distributed and accordingly each agent can determine their own control law using only the locally measured bearings. A convergence result is established in this paper which guarantees global asymptotic convergence of the formation to the desired formation shape.

**Relation to WP** This paper dicusses the problem of formation control of robots when they only have access to local bearing information. It provides an important theoretical result that could be of use when a system wants to make use of mobile sensors in the sensory layer. This work is associated with Task 3.1.

[1] http://top25.sciencedirect.com/subject/engineering/12/journal/systems-control-letters/01676911/archive/26/

# References

[1] Alper Aydemir, Adrian N. Bishop, and Patric Jensfelt. Simultaneous object class and pose estimation for mobile robotic applications with minimalistic recognition. In *Proc. of the IEEE International Conference on Robotics and Automation (ICRA'10)*, May 2010.

[2] Alper Aydemir, Kristoffer Sjöö, and Patric Jensfelt. Object search on a mobile robot using relational spatial information. In *Proc. of the 11th Int Conference on Intelligent Autonomous Systems (IAS-11)*, August 2010.

[3] M. Basiri, A.N. Bishop, and P. Jensfelt. Distributed control of triangular formations with angle-only constraints. *Systems & Control Letters*, pages 147–154, February 2010.

[4] Patrick Beeson, Matt MacMahon, Joseph Modayil, Aniket Murarka, Benajmin Kuipers, and Brian Stankiewicz. Integrating multiple representations of spatial knowledge for mapping, navigation, and communication. In *Interaction Challenges for Intelligent Assistants*, AAAI Spring Symposium, Stanford, CA, USA, 2007.

[5] Adrian N. Bishop and Patric Jensfelt. Global robot localization with random finite set statistics. In *Proc. of 13th International Conference on Information Fusion (FUSION 2010)*, Edinburgh, UK, July 2010.

[6] A.N. Bishop and P. Jensfelt. Stochastically convergent localization of objects by mobile sensors and actively controllable relative sensor-object pose. In *Proc. of ECC'09*, Budapest, Hungary, 2009.

[7] A.N. Bishop and P. Jensfelt. A stochastically stable solution to the problem of robocentric mapping. In *Proc. of ICRA'09*, Kobe, Japan, 2009.

[8] Anders Boberg, Adrian N. Bishop, and Patric Jensfelt. Robocentric mapping and localization in modified spherical coordinates with bearing measurements. In *Proc. of the Fifth International Conference on Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP 2009)*, December, Melbourne, Australia, 2009.

[9] Rodney A. Brooks. Visual map making for a mobile robot. In *Proc. of the IEEE International Conference on Robotics and Automation (ICRA'85)*, pages 824–829, 1985.

[10] José A. Castellanos, J. Neira, and Juan D. Tardós. Multisensor fusion for simultaneous localization and map building. *IEEE Transactions on Robotics and Automation*, 17(6):908–914, December 2001.

[11] Margarita Chli and Andrew J. Davison. Active matching. In *Proc. of European Conference on Computer Vision (ECCV'08)*, 2008.

[12] Kok Seng Chong and Lindsay Kleeman. Sonar based map building for a mobile robot. In *Proc. of the IEEE/RSJ International Conference on Robotics and Automation (ICRA'97)*, volume 2, pages 1700–1705, Albuquerque, New Mexico, April 1997. IEEE.

[13] Javier Civera, Andrew J. Davison, J. A. Magallón, and J. M. M. Montiel. Drift-free real-time sequential mosaicing. *International Journal of Computer Vision (IJCV)*, 2009.

[14] A.G. Cohn and S.M. Hazarika. Qualitative spatial representation and reasoning: An overview. *Fundamenta Informaticae*, 2001.

[15] M. Cummins and P. Newman. Highly scalable appearance-only SLAM–FAB-MAP 2.0. In *Proc. Robotics Science and Systems (RSS'09)*, 2009.

[16] Mark Cummins and Paul Newman. FAB-MAP: Probabilistic localization and mapping in the space of appearance. *The International Journal of Robotics Research (IJRR)*, 27(6), 2008.

[17] R. Davis, H. Shrobe, and P. Szolovits. What is a knowledge representation? *AI Magazine*, 14(1):17–33, 1993.

[18] Andrew J. Davison and David Murray. Simultaneous localisation and map-building using active vision. *IEEE Trans. PAMI*, 24(7):865–880, July 2002.

[19] Pantelis Elinas, Robert Sim, and James J. Little. $\sigma$SLAM: Slam: Stereo vision slam using the rao-blackwellised particle filter and a novel mixture proposal distribution. In *Proc. of the IEEE International Conference on Robotics and Automation (ICRA'06)*, Orlando, FL, May 2006.

[20] C. Galindo, A. Saffiotti, S. Coradeschi, P. Buschka, J.A. Fernández-Madrigal, and J. Gonález. Multi-hierarchical semantic maps for mobile robotics. In *Proc. of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'05)*, pages 3492–3497, 2005.

[21] Thomas David Garvey. *Perceptual strategies for purposive vision.* PhD thesis, Stanford, CA, USA, 1976.

[22] Patric Jensfelt and Henrik I. Christensen. Pose tracking using laser scanning and minimalistic environmental models. *IEEE Transactions on Robotics and Automation*, 17(2):138–147, April 2001.

[23] Patric Jensfelt, Danica Kragic, John Folkesson, and Mårten Björkman. A framework for vision based bearing only 3D SLAM. In *Proc. of*

*the IEEE International Conference on Robotics and Automation (ICRA'06)*, Orlando, FL, May 2006.

[24] Stephen Kaplan. *Environmental design research*, volume 1, chapter Cognitive maps, human needs and the designed environment. Dowden, Hutchinson and Ross, Stroudsburg, PA, 1973.

[25] David Kortenkamp. *Cognitive maps for mobile robots: A representation for mapping and navigation.* PhD thesis, University of Michigan, 1993.

[26] David Kortenkamp and Terry Weymouth. Topological mapping for mobile robots using a combination of sonar and vision sensing. In *Proc. of the National Conference on Artificial Intelligence (AAAI-94)*, 1994.

[27] B. Krieg-Bruckner, U. Frese, K. Luttich, C. Mandel, T. Mossakowski, and R. Ross. Specification of an Ontology for Route Graphs. In *Spatial Cognition IV: Reasoning, Action, Interaction. International Conference Spatial Cognition*, pages 390–412. Springer, 2004.

[28] B. J. Kuipers. *Spatial Orientation: Theory, Research, and Application*, chapter The cognitive map: Could it have been any other way?, pages 345–359. Plenum Press, New York, 1983.

[29] Benjamin Kuipers. Modeling spatial knowledge. *Cognitive Science*, 2:129–153, 1978.

[30] Benjamin Kuipers and Yung-Tai Byun. A robot exploration and mapping strategy based on a semantic hierachy of spatial representations. *Journal of Robotics and Autonomous Systems*, 8:47–63, 1991.

[31] Benjamin J. Kuipers. The spatial semantic hierarchy. *Artificial Intelligence*, 119:191–233, 2000.

[32] Benjamin Jack Kuipers. *Representing Knowledge of Large-Scale Space.* PhD thesis, MIT, Mathematical Department, Artificial Intelligence Laboratory, July 1977. TR-418.

[33] Thomas Lemaire, Simon Lacroix, and Joan Solà. A practical 3D bearing-only SLAM algorithm. In *Proc. of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'05)*, pages 2757–2762, August 2005.

[34] J. Leonard, H. Durrant-Whyte, and I.J. Cox. Dynamic map building for autonomous mobile robot. In *Proc. of the IEEE International Workshop on Intelligent Robots and Systems (IROS'90)*, volume 1, pages 89–96, July 1990.

[35] K. Lockwood, K. Forbus, D.T. Halstead, and J. Usher. Automatic categorization of spatial prepositions. In *Proceedings of the 28 th Annual Conference of the Cognitive Science Society.*, 2006.

[36] T.P. McNamara. Mental representations of spatial relations. *Cognitive Psychology*, 18:87–121, 1986.

[37] A. Pinz. Object categorization. *Foundations and Trends in Computer Graphics and Vision*, 1(4):255–353, 2006.

[38] A. Pronobis, B. Caputo, P. Jensfelt, and H. I. Christensen. A realistic benchmark for visual indoor place recognition. *Robotics and Autonomous Systems*, 58(1):81–96, January 2010.

[39] A. Pronobis, O. Martinez Mozos, and B. Caputo. SVM-based discriminative accumulation scheme for place recognition. In *Proc. of ICRA'08*, Pasadena, CA, USA, 2008.

[40] A. Pronobis, O. Martinez Mozos, B. Caputo, and P. Jensfelt. Multimodal semantic place classification. *The International Journal of Robotics Research (IJRR)*, 29(2-3):298320, February 2010.

[41] Andrzej Pronobis, Jie Luo, and Barbara Caputo. The more you learn, the less you store: Memory-controlled incremental SVM for visual place recognition. *Image and Vision Computing (IMAVIS)*, March 2010.

[42] Andrzej Pronobis, Kristoffer Sjöö, Alper Aydemir, Adrian N. Bishop, and Patric Jensfelt. A framework for robust cognitive spatial mapping. In *Proc. of the International Conference on Advanced Robotics (ICAR'09)*, Munich, Germany, June 2009.

[43] Andrzej Pronobis, Kristoffer Sjöö, Alper Aydemir, Adrian N. Bishop, and Patric Jensfelt. Representing spatial knowledge in mobile cognitive systems. In *11th International Conference on Intelligent Autonomous Systems (IAS-11)*, Ottawa, Canada, August 2010.

[44] Andrzej Pronobis, Kristoffer Sjöö, Alper Aydemir, Adrian N. Bishop, and Patric Jensfelt. Representing spatial knowledge in mobile cognitive systems. Technical Report TRITA-CSC-CV 2010:1 CVAP 316, Kungliga Tekniska högskolan (KTH), CVAP/CAS, Stockholm, Sweden, March 2010.

[45] Ananth Ranganathan and Frank Dellaert. Semantic modeling of places using objects. In *Proc. of Robotics: Science and Systems Conference (RSS07)*, 2007.

[46] T. Regier and L. A. Carlson. Grounding spatial language in perception: An empirical and computational investigation. *Journal of Experimental Psychology*, 130(2):273–2098, 2001.

[47] Robert Sim, Pantelis Elinas, Matt Griffin, and James J. Little. Vision-based slam using the rao-blackwellised particle filter. In *Proc. IJCAI Workshop on Reasoning with Uncertainty in Robotics*, Edinburgh, Scotland, July 30 2005.

[48] Kristoffer Sjöö, Alper Aydemir, and Patric Jensfelt. Mechanical support as a spatial abstraction for mobile robots. In *Proc. of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'10)*, October 2010.

[49] M. Skubic, D. Perzanowski, S. Blisard, A. Schultz, W. Adams, M. Bugajska, and D. Brock. Spatial language for human-robot dialogs. *IEEE Transactions on Systems, Man and Cybernetics*, 2(C-34):154–167, 2004.

[50] C. Theobalt, J. Bos, T. Chapman, A. Espinosa, M. Fraser, G. Hayes, E. Klein, T. Oka, and R. Reeve. Talking to godot: Dialogue with a mobile robot. In *Proc. of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'02)*, pages 1338–1343, 2002.

[51] Sebastian Thrun, Wolfram Burgard, and Dieter Fox. A real-time algorithm for mobile robot mapping with applications to multi-robot and 3D mapping. In *Proc. of the IEEE International Conference on Robotics and Automation (ICRA'00)*, volume 1, pages 321–328, San Francisco, CA, USA, 2000.

[52] Iwan Ulrich and Ilah Nourbakhsh. Appearance-based place recognition for topological localization. In *Proc. of the IEEE International Conference on Robotics and Automation (ICRA'00)*, volume 2, pages 1023–1029, April 2000.

[53] Shrihari Vasudevan, Stefan Gächter, Viet Nguyen, and Roland Siegwart. Cognitive maps for mobile robots – and object based approach. *Robotics and Autonomous Systems*, 55:359–371, 2007.

[54] Olle Wijk, Patric Jensfelt, and Henrik Christensen. Triangulation based fusion of ultrasonic sensor data. In *Proc. of the IEEE International Conference on Robotics and Automation (ICRA'98)*, volume 4, pages 3419–24, Leuven, Belgium, May 1998. IEEE.

[55] Jeremy L. Wyatt, Alper Aydemir, Michael Brenner, Marc Hanheide, Nick Hawes, Patric Jensfelt, Matej Kristan, Geert-Jan M. Kruijff, Pierre Lison, Andrzej Pronobis, Kristoffer Sjöö, Danijel Skočaj, Alen Vrečko, Hendrik Zender, and Michael Zillich. Self-understanding & self-extension: A systems and representational approach. *IEEE Transactions on autonomous mental development*, 2010. Accepted for publication.

[56] Hendrik Zender. Multi-layered conceptual spatial mapping. Technical report, DFKI GmbH, 2010.

[57] Hendrik Zender, Christopher Koppermann, Fai Greeve, and Geert-Jan M. Kruijff. Anchor-progression in spatially situated discourse: a production experiment. In *Proceedings of the Sixth International Natural Language Generation Conference (INLG 2010)*, pages 209–213, Trim, Co. Meath, Ireland, July 2010. Association for Computational Linguistics.

[58] Hendrik Zender, Geert-Jan M. Kruijff, and Ivana Kruijff-Korbayová. Situated resolution and generation of spatial referring expressions for robotic assistants. In *Proc. of JCAI'09*, 2009.

# Representing Spatial Knowledge in Mobile Cognitive Systems

Andrzej PRONOBIS [a], Kristoffer SJÖÖ [a], Alper AYDEMIR [a],
Adrian N. BISHOP [b], Patric JENSFELT [a]
[a] *Centre for Autonomous Systems, Royal Institute of Technology, Stockholm*
{pronobis, krsj, aydemir, patric}@kth.se
[b] *National ICT Australia (NICTA), Australian National University (ANU)*
adrian.bishop@nicta.com.au

**Abstract.** A cornerstone for cognitive mobile agents is to represent the vast body of knowledge about space in which they operate. In order to be robust and efficient, such representation must address requirements imposed on the integrated system as a whole, but also resulting from properties of its components. In this paper, we carefully analyze the problem and design a structure of a spatial knowledge representation for a cognitive mobile system. Our representation is layered and represents knowledge at different levels of abstraction. It deals with complex, cross-modal, spatial knowledge that is inherently uncertain and dynamic. Furthermore, it incorporates discrete symbols that facilitate communication with the user and components of a cognitive system. We present the structure of the representation and propose concrete instantiations.

## 1. Introduction

Many recent advances in the fields of robotics and artificial intelligence have been driven by the ultimate goal of creating artificial cognitive systems able to perform human-like tasks. Several attempts have been made to create integrated cognitive architectures and implement them on mobile robots [2,3,13,1,4]. There is an increasing interest in, and demand for, robots that are capable of dealing with complex and dynamic environments outside the traditional industrial workplaces. These next generation robots will not only have to track their position and navigate between points in space, but reason about space and their own knowledge, plan tasks and knowledge acquisition and interact with people in a natural way.

Spatial knowledge constitutes a fundamental component of the knowledge base of a cognitive agent providing a basis for navigation, reasoning, planning and episodic memories. Moreover, it is a common ground for communication between a robot and a human. In order for the process of acquisition, interpreting, storing and recalling of the spatial knowledge to be robust and efficient under limited resources and in realistic settings, the knowledge must be properly structured and represented. Such knowledge representation must address requirements imposed on the integrated system as a whole, but also resulting from properties of its components. Due to this central role, the design of a spatial knowledge representation should be one of the first steps in building a cognitive system.

In this work, we develop a structure of a spatial knowledge representation for a cognitive mobile system that we call COARSE (Cognitive lAyered Representation of Spatial knowledgE). We carefully analyze the role of a spatial represen-

tation and formulate design assumptions and requirements imposed by the functionality and components of an integrated system. Our representation is layered and represents knowledge at different levels of abstraction, from low-level sensory input to high level conceptual symbols. It is designed for representing complex, cross-modal, spatial knowledge that is inherently uncertain and dynamic and includes discrete symbols that facilitate communication with the user and components of the system. Moreover, we propose models and algorithms that could be used as instantiations of each layer of the representation.

This paper is motivated by the desire to create a framework that is powerful, robust and efficient, but most importantly suited for mobile agents performing typical human-like tasks. The literature contains many algorithms for spatial mapping and instantiations of mobile robotic systems. However, the existing representations are either designed for a very specific domain [7,12], they concentrate on a fraction of the spatial knowledge [20,23] or are designed to solve a single algorithmic task very efficiently rather than for use within a larger system [8,10,18]. The idea of this paper, is to take a step back, focus on structuring the whole body of spatial knowledge and see how an analysis of requirements can lead the way towards a powerful spatial representation for a cognitive mobile robot.

## 2. Related Work

There exists a broad literature on mobile robot localization, navigation and mapping and many algorithms relying on spatial knowledge have been proposed. These include solutions to such problems as Simultaneous Localization and Mapping (SLAM) [8,15,10,18] or place classification [16,20]. Every such algorithm maintains a representation of spatial knowledge. However, this representation is usually specific to the particular problem and designed to be efficient within the single mapping system detached from any other interacting components. Other, more general concepts, such as the Spatial Semantic Hierarchy [14] concentrate on lower levels of spatial knowledge abstraction and do not support higher-level conceptualization or representation of categorical information.

At the same time, we witness a growing interest in building artificial mobile cognitive systems [2,3,1,4]. These are complex, usually modular, systems that require a unified and integrated approach to spatial knowledge representation. The central role of spatial knowledge in those systems has been recognized and several authors proposed subsystems processing spatial knowledge integrated with other components such as dialogue systems [25,22]. However, neither of those provides a clear structure of the represented knowledge, perform a thorough analysis of the needs of different components of a mobile cognitive system or encapsulates all major aspects of spatial knowledge.

The most comprehensive relevant representation has been proposed in [25]. However, it has several major drawbacks that makes it unsuitable for systems that deal with dynamic and uncertain knowledge within large-scale, complex environments. First of all, the knowledge is never fully abstracted and is always grounded in an accurate global metric map. This makes the system less robust and scalable. Moreover, the categorical knowledge is not explicitly represented. The high-level conceptualization relies on rigid ontologies and ignores uncertainties associated

with represented symbols. Finally, it is modality-specific and does not allow for knowledge fusion from multiple sources. In the rest of the paper, we propose an approach to spatial knowledge representation that addresses those problems.

## 3. Analysis of the Problem

Before designing a representation of spatial knowledge, it is important to review the aspects a representation should focus on. In this section, we analyze those aspects and propose our definition of a generic spatial knowledge representation. Then, we formulate the problem within the context of cognitive systems.

### *3.1. What is a Spatial Knowledge Representation?*

Following the analysis by Davis [9], we formulate several points that characterize a general representation of spatial knowledge. A spatial representation can be seen as:

*a)* A substitution (surrogate) for the world that allows the agent to perform reasoning about the parts of the environment which are beyond its sensory horizon. Such a surrogate is naturally imperfect, and is incomplete (some aspects are not represented), inaccurate (captured with uncertainty), and will become invalid (e.g. due to dynamics of the world that cannot be observed and is too complex to be captured by the representation). Moreover, since the representation cannot be perfect, all the inferences based on that representation, such as the outcomes of the localization process, are uncertain. The only perfect representation of the world or the environment in which the agent operates is the environment itself.

*b)* A set of ontological commitments that determine the terms in which the agent thinks about space. The representation defines the aspects of the world that should be represented. Moreover, it defines the level of detail at which they should be represented as well as their persistence. The ontology should be understood in more general terms, from spatial concepts and their relations to categorical models or types of features extracted from the sensory input.

*c)* A set of definitions that determine the reasoning that can be (and that should be) performed within the framework and the possible inferences and their outcomes. The reasoning will typically correspond to determining the current location with respect to the internal map (topologically, semantically etc.), providing necessary knowledge for the navigation process, determining the properties of a location in space etc. Moreover, the representation defines how the location of the agent is represented and in what terms it is possible to refer to points in space (e.g. in terms of metric coordinates, semantic category of a place etc.).

*d)* A way of structuring the spatial information so that it is computationally feasible to perform all the necessary processing and inferences in a specified time (e.g. in real time) despite limited resources.

*e)* A medium of communication between the agent and human. If the agent is supposed to exchange information with humans, the representation must be designed in a way that allows the agent to interpret human expressions and generate expressions that are comprehensible to humans.

*f)* Similarly, a medium of communication between components of an integrated system.

### *3.2. Spatial Representation for Mobile Cognitive Systems*

In this work, we narrow the focus to mobile cognitive systems. Based on the analysis of existing approaches [3,1,23] as well as ongoing research on artificial cognitive systems [2], we have identified several areas of functionality, usually realized through separate subsystems, that must be supported by the representation. These include localization, navigation, and autonomous exploration, but also understanding and exploiting semantics associated with space, human-like conceptualization and categorization of space, reasoning about spatial units and their relations, human-robot communication, action planning, object finding and visual servoing, and finally recording and recalling episodic memories.

Having in mind the aforementioned functionalities, aspects covered by a representation of spatial knowledge as well as limitations resulting from practical implementations, we have identified several desired properties and designed a representation reflecting those properties.

Complex, cross-modal, spatial knowledge in realistic environments is inherently uncertain and dynamic. Therefore, it is futile to represent the environment as accurately as possible. A very accurate representation must be complex, require a substantial effort to synchronize with the world and still cannot guarantee that sound inferences will lead to correct conclusions [9]. Our primary assumption is that the representation should instead be minimal and inherently coarse and the spatial knowledge should be represented only as accurately as it is required to support the functionality of the system. Furthermore, redundancy should be avoided and whenever possible and affordable, new knowledge should should be inferred from the existing information. It is important to note that uncertainties associated with represented symbols should be explicitly modeled.

Information should be abstracted as much as possible to make it robust to dynamic changes. Moreover, representations that are more abstract should be used for longer-term storage. At the same time, knowledge extracted from immediate observations can be much more accurate (e.g. for the purpose of visual servoing). In other words, the agent should use the world as an accurate representation whenever possible. It is important to mention that rich and detailed representations should not constitute a permanent base for more abstract ones (as is the case in [25]). Similarly, space should be represented on different spatial scales from single scenes to whole environments.

Space should be discretized into a finite number of spatial units. Discretization of continuous space is one of the most important abstracting steps as it allows to make the representation robust, compact and tractable. Discretization drastically reduces the number of states that have to be considered e.g. during the planning process [11] and serves as a basis for higher level conceptualization [25].

A representation should allow not only for representing instantiations of spatial segments visited by the robot. It is equally important to provide means for representing unexplored space. Furthermore, categorical knowledge should be represented that is not specific to any particular location and instead corresponds to general knowledge about the world. Typical examples would be categorical models of appearance of places [20] or objects [19].

Finally, we focus on the fundamental role of the representation in human-robot interaction. Spatial knowledge representation should model correspondence

between the represented symbols and human concepts of space. Spatial properties (e.g. shape, size), semantic categories of rooms (e.g. kitchen, office) or spatial segments (e.g. rooms, floors, buildings) recognized by humans are examples of such concepts. This correspondence could be used to generate and resolve spatial referring expressions [24] or path descriptions.

## 4. Structure of the Representation

In this section, we propose a representation of spatial knowledge that adheres to the desired properties formulated above. Figure 1 gives a general overview of the structure of the representation. It is sub-divided into four layers which can be regarded as sub-representations focusing on different aspects of the world, abstraction levels of the spatial knowledge and different spatial scales. Moreover, each layer defines its own spatial entities and the way the agent's position in the world is represented. The properties of each layer are summarized in Table 1.

At the lowest abstraction level, we have the sensory layer which maintains an accurate representation of the robot's immediate environment extracted directly from the robot's sensory input. Higher, we have the place and categorical layers. The place layer provides fundamental discretisation of the continuous space into a set of distinct places. The categorical layer focuses on low-level, long-term categorical models of the robot's sensory information. Finally, at the top, we have the conceptual layer, which associates human concepts with the categorical models in the categorical layer and groups places into human-compatible spatial segments such as rooms. The following sections provide details about each of the layers.

### *4.1. Sensory Layer*

In the sensory layer, a detailed robocentric model of the robot's immediate environment is represented based on direct sensory input as well as data fusion over space around the robot and short time intervals. The sensory layer stores low-level features and landmarks extracted from the sensory input together with their exact position with respect to the robot. Measures of uncertainty are also included in this representation. Landmarks that move beyond a certain distance are forgotten and replaced by new information. Thus, this representation is akin to a sliding window, with robocentric and up-to-date direct perceptual information. It is also essentially bottom-up only, though directives and criteria, such as guiding the attentional process, may be imposed from upper layers.

The representation in the sensory layer helps to maintain stable and accurate information about the relative movements of the robot. Moreover, it allows for maintaining and tracking the position of various features while they are nearby. This can be useful for providing "virtual sensing" such as 360° laser scans based on short-term temporal sensory integration as well as generation of features based on spatial constellations of landmarks located outside the field of view of the sensor. Additionally, it could be used for temporal filtering of sensory input or providing robustness to occlusions. Finally, the sensory layer can provide the low level robotic movement systems with data for deriving basic control laws such as for obstacle avoidance or visual servoing.

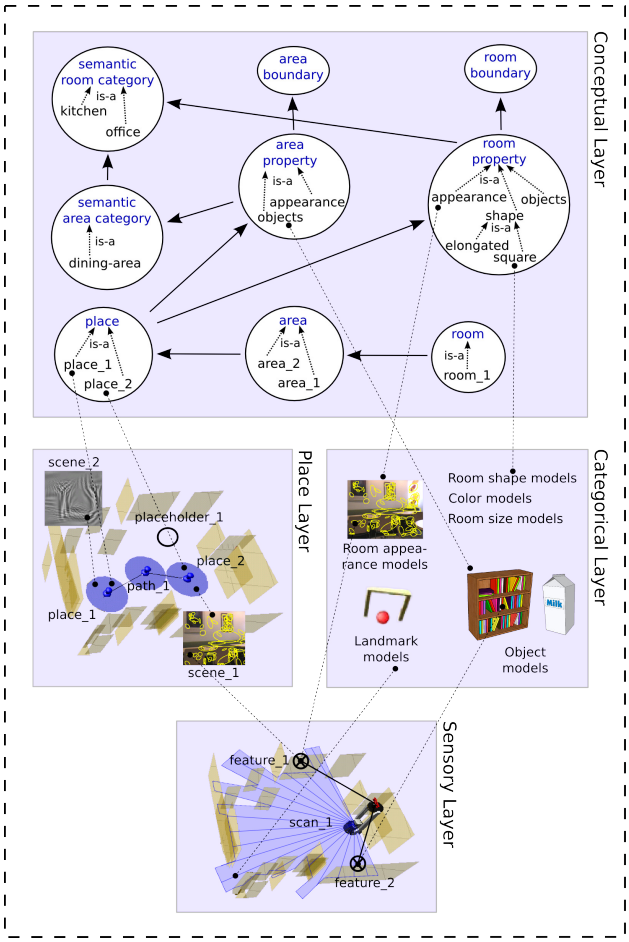



**Figure 1.** The layered structure of the spatial representation. The position of each layer within the representation corresponds to the level of abstraction of the spatial knowledge.

| Property | Sensory Layer | Place Layer | Categorical Layer | Conceptual Layer |
|---|---|---|---|---|
| Aspects represented | Accurate geometry and appearance | Local spatial relations, coarse appearance, geometry | Perceptual categorical knowledge | High-level spatial concepts / Links concepts ↔ entities |
| Agent's position | Pose within the local map | Place ID | Relationship to the categorical models | Expressed in terms of high level spatial concepts |
| Spatial scope | Small-scale, local | Large-scale | Global | Global |
| Knowledge persistence | Short-term | Long-term | Very long-term | Life-long / Very long-term |

**Table 1.** Comparison of properties of the four layers of the spatial representation.

### *4.2. Place Layer*

The place layer is responsible for the fundamental, bottom-up discretization of continuous space. In the place layer, the world is represented as a collection of basic spatial entities called places as well as their spatial relations. Each place is defined in terms of features that are represented in the sensory layer, but also spatial relations to other places. The aim of this representation is not to represent the world as accurately as possible, but at the level of accuracy sufficient for performing required actions and robust localization despite uncertainty and dynamic variations. Similarly, the relations do not have to be globally consistent as long as they are preserved locally with sufficient accuracy. The representation of places in the place layer persists over long term.

Besides places, the place layer also defines paths between them. The semantic significance of a path between two places is the possibility of moving directly between one and the other. This does not necessarily imply that the robot has traveled this path previously. A link might be created for unexplored place e.g. based on top-down cues resulting from the dialogue with the user (e.g. when the user indicates part of the environment that should be of interest to the robot, but not immediately). In addition, the place layer explicitly represents unexplored space. Tentative places are represented which the robot would probably uncover if it moved in a certain direction.

The place layer operates on distinct places as well as their connectivity and spatial relations to neighboring places. No global representation of the whole environment is maintained. Still, since the local connectivity is available, global representation (e.g. a global metric map) can be derived when needed. This representation will not be accurate, but will preserve the connectivity and relaxed spatial relations between all the places.

### *4.3. Categorical Layer*

The categorical layer contains long-term, low-level representations of categorical models of the robot's sensory information. The knowledge represented in this layer is not specific to any particular location in the environment. Instead, it represents a general long-term knowledge about the world at the sensory level. In this layer models of landmarks, objects or appearance-based room category or other properties of spatial segments such as shape, size or color are defined in terms of low-level features. The position of this layer in the spatial representation reflects the assumption that the ability to categorize and group sensory observations is the most fundamental one and can be performed in a feed-forward manner without any need for higher-level feedback from cognitive processes.

The categorical models stored in this layer give rise to properties that are utilized by conceptual layer. In many cases, the values of those properties will correspond to human spatial concepts, not to internal concepts of the robot (e.g. office-like appearance or elongated shape). The properties might require complicated models that can only be inferred from training data samples. In case of models that correspond to human concepts, they can be learned in a supervised fashion, using a top-down supervision signal.

### *4.4. Conceptual Layer*

The conceptual layer provides an ontology that represents taxonomy of the spatial concepts and properties of spatial entities that are linked to the low-level categorical models stored in the categorical layer. This associates semantic interpretations with the low-level models and can be used to specify which properties are meaningful e.g. from the point of view of human-robot interaction. Moreover, the conceptual layer represents relations between the concepts and instances of those concepts linked to the spatial entities represented in the place layer. This makes the layer central for verbalization of spatial knowledge and interpreting and disambiguating verbal expressions referring to spatial entities.

The second important role of the conceptual layer is to provide definitions of the spatial concepts related to the semantic segmentation of space based on the properties of segments observed in the environment. A building, floor, room or area are examples of such concepts. The conceptual layer contains information that floors are usually separated by staircases or elevators and that rooms usually share the same general appearance and are separated by doorways. Those definitions can be either given or learned based on asserted knowledge about the structure of a training environment introduced to the system.

Finally, the conceptual layer provides definitions of semantic categories of segments of space (e.g. rooms) in terms of values of properties of those segments. The properties can reflect the general appearance of a segment as observed from a place, its geometrical features or objects that are likely to be found in that place.

## 5. Instantiations

This section indicates specific models and algorithms maintaining those models that we propose to use for representing knowledge stored in each layer.

We propose to realize the sensory layer using a robocentric, metric SLAM [6, 5]. Robocentric mapping reflects the properties of the sensory layer and allows for a straightforward treatment of forgetting knowledge that falls outside a certain horizon around the robot. The robocentric map can be seen as a sliding window centered on the robot and containing a detailed view of the world, which allows the robot to maintain a drift free estimate of the pose as long as it stays in a local region of space. The SLAM algorithm explicitly represents the uncertainty associated with the pose of the robot and the location of all landmarks in the local surrounding using a multivariate Gaussian distribution [6,5].

We propose to instantiate the place layer based on the mapping framework proposed in [21]. Central to the approach is the place map represented as a collection of places. A place is defined by a subset of values of arbitrary, possibly complex, distinctive features and spatial relations reflecting the structure of the environment. The features provide information about the world and can be perceived by an agent when at that place. In this sense, the places build on the perception of the agent and are based on its perceptual capabilities.

The categorical layer can be seen as an ensemble of categorical models of the robot's sensory information. The literature provides a broad range of models that could be used for this purpose. First, in order to represent visual and geometrical

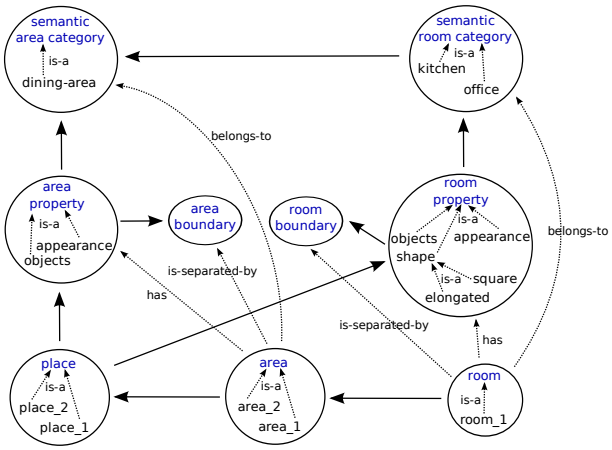



**Figure 2.** Overview of a possible instantiation of the conceptual layer. The solid arrows represent dependencies, while the dashed arrows illustrate the ontology that represents the taxonomy of spatial concepts and properties of spatial entities.

properties of areas in the environment, we suggest to use the multi-modal place classification algorithm presented in [20]. Other methods can be employed for representing landmarks (e.g. doors [17]) and object categories [19].

For the conceptual layer, we propose a possible instantiation presented in Figure 2. The conceptual layer provides an ontology that represents the taxonomy of the spatial concepts and properties as well as dependencies between the concepts, properties and instances of spatial entities. We use a fixed, handcrafted ontology for representing the taxonomy and a probabilistic model for representing the dependencies. In such an approach, the ontology is largely encoded in the structure of the probabilistic model. We represent the location of the robot within segments of space (e.g. a room or an area such as a dining area), the observed properties of areas and rooms as well as semantic categories of areas and rooms in terms of random variables. In the illustration in Figure 2, we can consider the circles as random variables and the solid arrows as dependencies within a graphical model. At the same time, the *is-a* relations link the random variables with their values. Further, the model represents the spatial hierarchy of segments of space. There is a dependency between the location of the robot at different levels of this hierarchy (e.g. a room and an area within the room). Moreover, the dependency between the instance of a place and the properties of areas and rooms observed from this place is represented. Those properties in turn influence the semantic categories of areas or rooms to which the place belongs. Finally, the proposed model represents the dependency between the area and room properties observed as the robot explores the environment and the probability that the robot crossed a boundary of a spatial segment. This link effectively defines the concepts of a room and an area and can be used to provide semantic segmentation of space.

## 6. Conclusions and Future Works

In this paper, we presented an analysis of the requirements for a spatial knowledge representation for cognitive systems and proposed a layered representation that conforms to those requirements. The representation provides a unified and coherent view on the structure of spatial knowledge and a basis for designing

artificial cognitive systems. We further proposed specific models and algorithms as possible instantiations. Future work will focus on integrating those algorithms, which so far were only evaluated in separation, into a complete spatial subsystem providing spatial understanding capabilities for a mobile robot.

## References


[1] COGNIRON: The Cognitive Robot Companion. Website: `www.cogniron.org`.
[2] CogX: Cognitive Systems that Self-Understand and Self-Extend. Website: `www.cogx.eu`.
[3] CoSy: Cognitive Systems for Cognitive Assistants. Website: `www.cognitivesystems.org`.
[4] RobotCub. Website: `www.robotcub.org`.
[5] A. N. Bishop and P. Jensfelt. A stochastically stable solution to the problem of robocentric mapping. In *Proc. of ICRA'09*.
[6] J. Castellanos, R. Martinez-Cantin, J. Tardos, and J. Neira. Robocentric map joining: Improving the consistency of EKF-SLAM. *Robotics and Autonomous Sys.*, 55(1), 2007.
[7] A. Chella and I. Macaluso. The perception loop in cicerobot, a museum guide robot. *Neurocomputing*, 72(4-6), 2009.
[8] M. Cummins and P. Newman. Highly scalable appearance-only SLAM - FAB-MAP 2.0. In *Proceedings of Robotics: Science and Systems V*, June 2009.
[9] R. Davis, H. Shrobe, and P. Szolovits. What is a knowledge representation? *AI Magazine*, 14(1):17–33, 1993.
[10] U. Frese and L. Schrder. Closing a million-landmarks loop. In *Proc. of IROS'06*.
[11] N. Hawes, H. Zender, K. Sjöö, M. Brenner, G.-J. M. Kruijff, and P. Jensfelt. Planning and acting with an integrated sense of space. In *Proc. of HYCAS'09*.
[12] P. Jensfelt, E. Förell, and P. Ljunggren. Automating the marking process for exhibitions and fairs. *Robotics and Autonomous Magazine*, 14(3):35–42, Sept. 2007.
[13] G.-J. Kruijff, H. Zender, P. Jensfelt, and H. Christensen. Situated dialogue and spatial organization: What, where and why? *Int. Journ. of Advanced Robotic Systems*, 4(1), 2007.
[14] B. Kuipers. The spatial semantic hierarchy. *Artificial Intelligence*, 119:191–233, 2000.
[15] M. Milford and G. Wyeth. Mapping a suburb with a single camera using a biologically inspired slam system. *IEEE Transactions on Robotics*, 24(5):1038–1053, Oct. 2008.
[16] O. M. Mozos, R. Triebel, P. Jensfelt, A. Rottmann, and W. Burgard. Supervised semantic labeling of places using information extracted from laser and vision sensor data. *Robotics and Autonomous Systems*, 55(5):391–402, May 2007.
[17] A. Murillo, J. Kosecka, J. Guerrero, and C. Sagues. Visual door detection integrating appearance and shape cues. *Robotics and Autonomous Systems*, 56(6), 2008.
[18] L. Paz, P. Jensfelt, J. Tardós, and J. Neira. EKF SLAM updates in O(n) with Divide and Conquer SLAM. In *Proc. of ICRA'07*.
[19] A. Pinz. Object categorization. *Foundations and Trends in Computer Graphics and Vision*, 1(4):255–353, 2006.
[20] A. Pronobis, O. M. Mozos, B. Caputo, and P. Jensfelt. Multi-modal semantic place classification. *The International Journal of Robotics Research*, 29(2-3), 2010.
[21] A. Pronobis, K. Sjöö, A. Aydemir, A. N. Bishop, and P. Jensfelt. A framework for robust cognitive spatial mapping. In *Proc of ICAR'09*.
[22] T. Spexard, S. Li, B. Wrede, J. Fritsch, G. Sagerer, O. Booij, Z. Zivkovic, B. Terwijn, and B. Krose. BIRON, where are you? enabling a robot to learn new places in a real home environment by integrating spoken dialog and visual localization. In *Proc. of IROS'06*.
[23] S. Thrun, A. Bücken, W. Burgard, D. Fox, T. Fröhlinghaus, D. Henning, T. Hofmann, M. Krell, and T. Schmidt. Map learning and high-speed navigation in RHINO. In D. Kortenkamp, R. Bonasso, and R. Murphy, editors, *AI-based Mobile Robots: Case Studies of Successful Robot Systems*. MIT Press, 1998.
[24] H. Zender, G.-J. M. Kruijff, and I. Kruijff-Korbayová. Situated resolution and generation of spatial referring expressions for robotic assistants. In *Proc. of JCAI'09*.
[25] H. Zender, O. M. Mozos, P. Jensfelt, G.-J. M. Kruijff, and W. Burgard. Conceptual spatial representations for indoor mobile robots. *Robotics and Autonomous Systems*, 56(6), 2008.

# A Framework for Robust Cognitive Spatial Mapping

Andrzej Pronobis, Kristoffer Sjöö, Alper Aydemir, Adrian N. Bishop and Patric Jensfelt

***Abstract*— Spatial knowledge constitutes a fundamental component of the knowledge base of a cognitive, mobile agent. This paper introduces a rigorously defined framework for building a cognitive spatial map that permits high level reasoning about space along with robust navigation and localization. Our framework builds on the concepts of *places* and *scenes* expressed in terms of arbitrary, possibly complex features as well as local spatial relations. The resulting map is topological and discrete, robocentric and specific to the agent's perception. We analyze spatial mapping design mechanics in order to obtain rules for how to define the map components and attempt to prove that if certain design rules are obeyed then certain map properties are guaranteed to be realized. The idea of this paper is to take a step back from existing algorithms and literature and see how a rigorous formal treatment can lead the way towards a powerful spatial representation for localization and navigation. We illustrate the power of our analysis and motivate our cognitive mapping characteristics with some illustrative examples.**

## I. Introduction

An autonomous mobile agent needs to represent its surroundings in order to reason an plan actions within it. The typical spatial knowledge representations used in mobile robotics are purely metrical and rely on information extracted from simple, but accurate metric sensors. However, as the robots are designed to perform human-like tasks in more and more complex and dynamic environments [3], [8], [14], metrical global maps become harder to control and observe [5]. Moreover, it is not clear that the level of detail offered by such maps is necessary, or even desirable, when the agent is a cognitive system intended to interact with the world in a human-like way [5], [14]. It is commonly accepted [5], [8], [9], [14], that the spatial knowledge of a cognitive agent should be abstracted in order to make it robust to dynamic variations, easier to maintain and useful for spatial reasoning. At the same time, the agent should be able to exploit sensory information that might be complex and non-metric [3], [8], [9], yet reflects crucial aspects of the environment.

This paper is motivated by desire to create a powerful cognitive mapping framework, which is suitable for cognitive conceptualization, encompasses complex spatial information, and provides robustness against natural changes in the environment, while maintaining a description that permits formal proofs and derivations. Although the literature contains many algorithms for spatial mapping, there is little work on the formal analysis of their fundamental requirements and properties. The idea of this paper, is to take a step back and see how a rigorous formal treatment can lead the way towards a powerful spatial representation for localization and navigation.

The authors are with the Centre for Autonomous Systems at the Royal Institute of Technology (KTH), Stockholm, Sweden. This work was supported by the Centre for Autonomous Systems (CAS) and the EU FP7 project CogX and the Swedish Research Council, contract 621-2006-4520 (K. Sjöö) and 2005-3600-Complex (A. Pronobis).

The contribution of the work presented here is a cognitive mapping framework that builds on the concepts of *places* and *scenes* expressed in terms of arbitrary, possibly complex features as well as local spatial relations. The resulting map is topological and discrete, robocentric and specific to the agent's perception. We analyze spatial mapping design mechanics in order to obtain rules for how to define the map components and attempt to prove that if certain design rules are obeyed then certain map properties are guaranteed to be realized. Moreover, we suggest localization and navigation strategies that can be applied in this framework. Finally, we illustrate the power of our analysis and motivate our cognitive mapping characteristics with illustrative examples.

The paper is organized as follows: after a general overview of the framework, Section III presents the formal definition of the map and its components. Then, Section IV gives a method for expressing the map through a set of functions and provides rules that must be obeyed in order for the map to be valid. Sections V and VI propose methods for performing navigation as well as probabilistic localization within the framework. The paper concludes with a summary and a brief discussion.

## II. An Overview of the Framework

The role of a cognitive map is not to represent the world as accurately as possible, but rather to allow the agent to act in an environment despite uncertainty and dynamic variations. Such a map does not need to provide perfect global consistency as long as the local spatial relations are preserved with sufficient accuracy. In our framework, the map is represented as a collection of basic spatial entities called *places*.

A *place* is defined by a subset of values of arbitrary, possibly complex, distinctive features and spatial relations reflecting the structure of the environment. The features provide information about the world and can be perceived by an agent when at that place. In this sense, the places build on the perception of the agent and are based on its perceptual capabilities. Additionally, we introduce the concept of a *scene* which facilitates the generation of places by providing groupings of similar feature values. In addition to this, a scene provides a segmentation of space that serves as a basis for defining spatial relations.

The structure of the framework and its formalization described in the next section represent a certain view on

a cognitive map. First, the map is defined in terms of the agent's perception of space and adapts to its perceptual capabilities. Second, the perceived features can be abstract and non-metric and describe for instance visual properties of the world. In this sense, the map is subjective and robocentric as the robot's observations do not have to be expressed in terms of any objectively defined quantities or any global coordinate system. The map is fragmented (consists of a set of independent places), topological and does not require maintaining global spatial consistency.

This framework is designed so that a robot can build from the bottom-up a cognitive map of the environment which follows certain cognitive principles. The idea is that such principles can actually lead to better performance in localization, navigation and loop-closing for robots moving in large-scale environments; e.g. see the practical demonstrations in [5], [8]. The work of [5] involves a similarly designed mapping framework to the one analyzed in this paper and motivates the need to take a step back and analyze what desirable properties of the cognitive map can be provably obtained. The next section provides a formal definition of the place map and each of its components.

## III. Definition of the Place Map

Consider a set $\{f_i\}_{i=1}^{n_f}$ of *features* $f_i$ defined as

$$f_i\left(\boldsymbol{x}, t\right) : \mathcal{C} \times \mathbb{R} \rightarrow \mathcal{F}_i \in \mathbb{R}^n \tag{1}$$

where $\mathcal{C}$ represents the *configuration space* of the agent (e.g. $\mathcal{C} = \mathbb{R}^2$ if only position in a 2D metric space is considered and $\mathcal{C} = \mathbb{R}^2 \times SO(1)$ if the value of features can depend on both position and heading), $t \in \mathbb{R}$ represents time, and $\mathcal{F}_i$ is the range of values of the feature $f_i$. Features thus provide information about the world as it would be perceived by an agent located at the configuration $\boldsymbol{x} \in \mathcal{C}$. Each feature can be time-varying.

An example feature-type is Euclidean distance, $f_i(\boldsymbol{x}, t) = \|\boldsymbol{x} - \boldsymbol{y}\|_2$, with $\mathcal{F}_i = [0, \infty)$, which maps every point in $\mathcal{C}$ to a value dependent on how far $\boldsymbol{x} \in \mathcal{C}$ is to a specific landmark located at $\boldsymbol{y} \in \mathcal{C}$. Features do not necessarily have to describe metric properties of the world (such as distance or size). Consider for instance a visibility-type feature for which $\mathcal{F}_i = \{0, 1\}$, which relates every pose $\boldsymbol{x} \in \mathcal{C}$ to a binary output depending on whether or not a specific landmark is visible in that pose. Another example would be a feature $f_i(\boldsymbol{x}, t)$ with $\mathcal{F}_i = [0, 1]$, which represents the average hue perceived by the robot's visual sensor, or even the full HSV color space, in which case $\mathcal{F}_i = [0, 1] \times [0, 1] \times [0, 1]$. Such features may be time-varying e.g. due to changes in illumination.

Other, more abstract, feature types are possible in this framework. An example could be features typically employed in visual topological localization [3], [5], [9] such as clouds of image keypoints characterized by the local SIFT [7] or SURF [2] descriptors. Features such as the "gist" of a scene [13] (principal components of outputs of spatially oriented image filters) or other global image features applied for visual place classification [9] could also be used in this framework in a straightforward manner. In such case, $f_i(\boldsymbol{x}, t)$ is a vector representing local descriptors for the $N$ strongest keypoints or the global descriptor.

Given the definition of features, we can now introduce the *feature space*

$$\mathcal{F} = \mathcal{F}_1 \times \mathcal{F}_2 \times \ldots \times \mathcal{F}_{n_f}, \tag{2}$$

in which each tuple $(\zeta_1, \ldots, \zeta_{n_f})$ of the feature values $\zeta_i = f_i(\boldsymbol{x}, t)$ corresponds to a single point. We are now ready to define the concept of a *scene*.

*Definition 1:* We introduce a set $\{\mathcal{S}_i\}_{i=1}^{n_s}$ of *scenes* $\mathcal{S}_i$ defined as

$$\mathcal{S}_i = \left\{(\zeta_1, \ldots, \zeta_{n_f})\right\} \subseteq \mathcal{F} \tag{3}$$

such that $\forall_i \mathcal{S}_i \neq \emptyset$ and $\forall_{i \neq j} \mathcal{S}_i \cap \mathcal{S}_j = \emptyset$. In other words, scenes are (non-overlapping) collections of tuples of features that could be perceived by the robot. Then, it is possible to specify the extent of a scene in the configuration space at time $t$:

$$\mathcal{C}_{\mathcal{S}_i}(t) = \left\{\boldsymbol{x} \in \mathcal{C} : \left(f_1(\boldsymbol{x}, t), \ldots, f_{n_f}(\boldsymbol{x}, t)\right) \in \mathcal{S}_i\right\} \tag{4}$$

It is important to note that no assumptions need to be made about the properties or structure of the feature functions in order to determine if a point $\boldsymbol{x} \in \mathcal{C}$ is within the spatial extent of a scene $\mathcal{S}_i$ at time $t$. In particular, a closed-form expression is not required as long as the feature values can be obtained. This has important practical implications as it permits the use of more complex features.

The definition of scenes gives raise to a segmentation of the configuration space. Depending on the features, however, this segmentation may not reflect the spatial relationships in the world that constitute a large portion of the spatial knowledge. Intuitively, the definition of scenes leads to a division of metric space into regions based on properties such as appearance. As such, two distant disconnected regions could share similar properties (see e.g. Figure 1(a)). Additional power to distinguish between such regions can be attained using knowledge about spatially neighboring regions.

Consider a set $\{r_i\}_{i=1}^{n_r}$ of *spatial relations* $r_i$ defined as

$$r_i\left(\boldsymbol{x}, t\right) : \mathcal{C} \times \mathbb{R} \rightarrow \mathcal{R}_i \in \mathbb{R}^m, \tag{5}$$

where $\mathcal{R}_i$ is the range of values of the relation $r_i$. Each *spatial relation* $r_i$ is defined with respect to the set of scenes $\{\mathcal{S}_i\}_i$ and describes the spatial relation of the point $\boldsymbol{x}$ in the configuration space $\mathcal{C}$ at time $t$ to some or all of those scenes. Relations permit discriminating between points using region-based concepts such as the region connection calculus, RCC [10], often applied in qualitative spatial reasoning. Moreover, in many cases, the values of relations can be estimated in practice by performing a dynamic action in the environment (e.g. the agent moving between points in configuration space that correspond to different scenes).

Consider, for instance, the adjacency relation for which $\mathcal{R}_i = \{1, 0\}$. The adjacency relation $r_{\mathcal{S}_i}(\boldsymbol{x}, t)$ of a point $\boldsymbol{x} \in \mathcal{C}$ to the region $\mathcal{C}_{\mathcal{S}_i}$ can be expressed in terms of the RCC-8 [10] predicate $\mathsf{EC}$ (externally connected) as

$$r_{\mathcal{S}_i}(\boldsymbol{x}, t) = \bigvee_{\mathcal{S}_j} \boldsymbol{x} \in \mathcal{C}_{\mathcal{S}_j} \wedge \mathsf{EC}(\mathcal{C}_{\mathcal{S}_i}, \mathcal{C}_{\mathcal{S}_j}). \tag{6}$$

Alternatively, a relation could be defined based on the minimum distance between a region $\mathcal{C}_{\mathcal{S}_i}$ and a point $\boldsymbol{x}$ in the configuration space as follows

$$r_{\mathcal{S}_i}(\boldsymbol{x}, t) = \min_{\boldsymbol{y} \in \mathcal{C}_{\mathcal{S}_i}} \|\boldsymbol{x} - \boldsymbol{y}\|. \tag{7}$$

We have now defined scenes and spatial relations, the main building blocks of the spatial entities constituting our map. Analogously to the feature space, we can introduce the *place descriptor space*

$$\mathcal{D} = \mathcal{F} \times \mathcal{R}_1 \times \mathcal{R}_2 \times \ldots \times \mathcal{R}_{n_r}, \tag{8}$$

in which each tuple $D = (\zeta_1, \ldots, \zeta_{n_f}, \rho_1, \ldots, \rho_{n_r})$ of the feature values and relation values $\rho_i = r_i(\boldsymbol{x}, t)$ corresponds to a single point.

*Definition 2:* Let us define the *place map* as a set

$$\mathcal{M} = \{\mathcal{P}_1, \mathcal{P}_2, \ldots, \mathcal{P}_{n_p}\} \tag{9}$$

of *places* $\mathcal{P}_i$ defined as

$$\mathcal{P}_i = \{D\} \subseteq \mathcal{D} \tag{10}$$

such that $\forall_i \mathcal{P}_i \neq \emptyset$ and $\forall_{i \neq j} \mathcal{P}_i \cap \mathcal{P}_j = \emptyset$.
In other words, similarly to scenes, places are groups of values of features; however, they encompass additional knowledge about the structure of the world encoded in the values of relations.

As a result, it is possible to specify the extent of a place in the configuration space at time $t$, as follows

$$\begin{aligned} \mathcal{C}_{\mathcal{P}_i}(t) \quad = \quad & \{\boldsymbol{x} \in \mathcal{C} : (f_1(\boldsymbol{x}, t), \ldots, f_{n_f}(\boldsymbol{x}, t), \\ & r_1(\boldsymbol{x}, t), \ldots, r_{n_f}(\boldsymbol{x}, t)) \in \mathcal{P}_i\} \end{aligned} \tag{11}$$

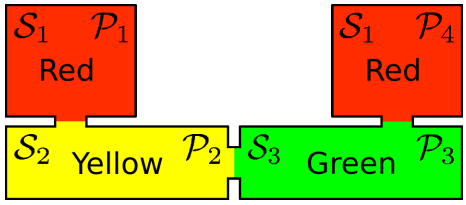
Note that not every point $\boldsymbol{x} \in \mathcal{C}$ is necessarily assigned to a place $\mathcal{P}_i$. The set of points $\mathcal{Q}(t) = \mathcal{C} \,/\, \bigcup\{\mathcal{C}_{\mathcal{P}_i}(t)\}_{i=1}^{n_p}$ is denoted *unassigned space* at time $t$. Again, no assumptions have to be made about the structure of the functions used to obtain the values of features and relations in order to determine if a point $\boldsymbol{x} \in \mathcal{C}$ is within the spatial extent of a place at time $t$.

Let us discuss the properties of places in the configuration space. Places are defined exclusively in terms of the values of features and spatial relations that are in functional relation to $(\boldsymbol{x} \in \mathcal{C}, t \in \mathbb{R})$. Moreover, places do not overlap in the descriptor space. As a consequence, places do not overlap in configuration space: $\forall_{i \neq j, t \in \mathbb{R}} \mathcal{C}_{\mathcal{P}_i}(t) \cap \mathcal{C}_{\mathcal{P}_j}(t) = \emptyset$.

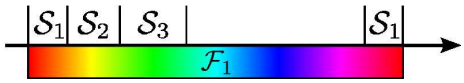
Also, if features and relations are time-invariant, the extents of places will be time-invariant as well. Typically, the nature of relations will mean that they are time-invariant as long as the features are. Note that if the configuration space reflects both position and heading, the places might spread across several positions and only a subset of headings.

### A. Example 1 - Abstract Features and Relations

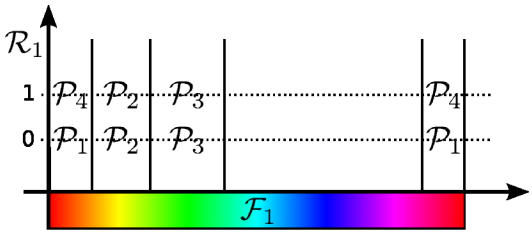
Consider a simple example of a small environment presented in Figure 1(a) consisting of 4 rooms characterized by the color of the floor. We define a single feature $f_1(\boldsymbol{x}, t) : \mathcal{C} \times \mathbb{R} \rightarrow \mathcal{F}_1$ that corresponds to the hue of the floor color at the location $\boldsymbol{x} \in \mathcal{C} = \mathbb{R}^2$. Then, the feature space is simply




(a) Map of the environment and metric extents of places.




(b) Scenes defined in the feature space.




(c) Places defined in the descriptor space.

Fig. 1. Illustrative example of an environment and definitions of places in the descriptor space.

defined by the range of the hue values e.g. $\mathcal{F}_1 = [0, 255]$. If we divide the feature space into regions as presented in Figure 1(b), we can differentiate between three scenes: red ($\mathcal{S}_1$), yellow ($\mathcal{S}_2$) and green ($\mathcal{S}_3$). We can clearly see that the scene $\mathcal{S}_1$ corresponds to two separate rooms which could be distinguished if we consider their relations to other scenes. Let us define an adjacency relation with respect to the scene $\mathcal{S}_3$, $r_1(\boldsymbol{x}, t) : \mathcal{C} \times \mathbb{R} \rightarrow \mathcal{R}_1 = \{1, 0\}$ as explained in Section III, and create the place descriptor space $\mathcal{D} = \mathcal{F}_1 \times \mathcal{R}_1$. In that space, we can create four non-overlapping places $\mathcal{P}_1$-$\mathcal{P}_4$ by dividing the scene $\mathcal{S}_1$ into two places, one of which is adjacent to the scene $\mathcal{S}_3$ and the other is not. This division is reflected in the clustering of the descriptor space presented in Figure 1(c).

## IV. Space Segmentation Using Applicability

The division of the feature space and descriptor space that gives rise to scenes and then to places can be expressed in many different ways. This section describes the segmentation in terms of real-valued functions over space, which encode the degree of belonging to the different places or scenes.

This view imposes certain restrictions on the functions and thereby on the features and relations used, but given that these are satisfied it is shown that a consistent segmentation results. As will be demonstrated in Sections V and VI, this information can also be used to support both navigation and localization. We describe these functions both for scenes and places, denoting the feature/descriptor space (as the case may be) by $\mathcal{A}$, and an arbitrary point in that space by $A$. The reasoning is analogous for both cases.

We introduce a set $\{g_i\}_{i=1}^{n_g}$ of *applicability functions* $g_i$ defined as

$$g_i(A) : \mathcal{A} \rightarrow \mathcal{G}_i \subseteq (\mathbb{R}^+ \cup \{0\}), \tag{12}$$

*Definition 3:* Given the set of applicability functions $\{g_i\}_{i=1}^{n_g}$, we define a cluster $\mathcal{K}_i \subseteq \mathcal{A}$ as

$$\mathcal{K}_i = \{A \in \mathcal{A} : g_i(A) > g_j(A) > 0, \forall i \neq j\} \quad (13)$$

and note especially that $g_i(A) = 0 \Rightarrow A \notin \mathcal{K}_i$.
Definition 3 suggests that we can think of the functions $g_i(A)$ as *measures* of how applicable a point $A$ is to the cluster $\mathcal{K}_i$. The clusters are non-overlapping in $\mathcal{A}$: $\forall_{i \neq j} \mathcal{K}_i \cap \mathcal{K}_j = \emptyset$. We now examine the requirements this places on the spatially defined feature and relation functions.

To do this let us introduce an additional function

$$\chi_i(\boldsymbol{x}) = g_i(A) = g_i(a_1(\boldsymbol{x}, t), \ldots, a_{n_a}(\boldsymbol{x}, t)) \quad (14)$$

which represents the applicability over the configuration space. (Here, the $a_i$ may be features only or features and relations, depending on whether $\mathcal{A} = \mathcal{F}$ or $\mathcal{A} = \mathcal{D}$.) As a result, it is similarly possible to specify the extent of a place in the configuration space at time $t$, as follows

$$\begin{aligned} \mathcal{C}_{\mathcal{P}_i}(t) &= \{\boldsymbol{x} \in \mathcal{C} : \chi_i(\boldsymbol{x}) > \chi_j(\boldsymbol{x}) > 0,\ \forall i \neq j\} \\ Q(t) &= \{\boldsymbol{x} : \chi_i(\boldsymbol{x}) = 0, \forall i\} \end{aligned} \quad (15)$$

However, this leaves parts of $\mathcal{C}$ undefined wherever no $\chi_i$ is greater than any other. If this occurs anywhere but on an infinitesimal borderline between places/scenes, it represents an ambiguity. To avoid this we introduce the following:

*Definition 4:* Let $\mu(\mathcal{S}) \geq 0$ denote the Lebesgue measure of the set $\mathcal{S}$ and $\Delta$ the set of all points not defined by Eq. 15. If $\mu(\Delta) = 0$, the spatial segmentation by $\{\chi_i\}$ is said to be *consistent*.

*Proposition 1:* Suppose that $\chi_i$ is a *piecewise analytical* function, i.e. that $\chi_i = \{\chi_{i,\alpha},\ \text{if } \mathbf{x} \in \mathbb{D}_{i,\alpha}\}, \forall \alpha$ where $\alpha$ is a countable index and where each $\chi_{i,\alpha}$ is a real analytic function on its open domain $\mathbb{D}_{i,\alpha}$ for all $t$. Assume that $\mu(\mathbb{D}_{i,\alpha}) > 0$ and $\{\bigcup_\alpha \mathrm{cl}(\mathbb{D}_{i,\alpha})\} = \mathcal{C}$ where $\mathrm{cl}(\mathbb{D}_{i,\alpha})$ is the closure of $\mathbb{D}_{i,\alpha}$ in $\mathcal{C}$. In the same way, let $\chi_j = \{\chi_{j,\beta},\ \text{if } \mathbf{x} \in \mathbb{D}_{j,\beta}\}, \forall \beta$ in the same way. Now assume that $\chi_i$ and $\chi_j$ are not identical on any entire intersection of their analytical pieces (except where both are identically zero):

$$\begin{aligned} &\forall \alpha \forall \beta : \mathbb{D}_{i,\alpha} \cap \mathbb{D}_{j,\beta} \neq \emptyset \Rightarrow \\ \Rightarrow\ &\chi_i(\mathbf{x}) \not\equiv \chi_j(\mathbf{x}) \vee \chi_i(\mathbf{x}) \equiv \chi_j(\mathbf{x}) \equiv 0 \text{ on } \mathbb{D}_{i,\alpha} \cap \mathbb{D}_{j,\beta} \end{aligned}$$

If the above holds for all pairs $i \neq j$, the segmentation of space into place via Eq. 15 is consistent, as per Definition 4.

*Proof:* The function $\chi_i - \chi_j$, is real and analytic on each non-empty $\mathbb{D}_{ij,\alpha,\beta} \triangleq \mathbb{D}_{i,\alpha} \cap \mathbb{D}_{j,\beta}$ Because of this, on $\mathbb{D}_{ij,\alpha,\beta}$ the zeros of $\chi_i - \chi_j$ are isolated unless $\chi_i$ and $\chi_j$ are equivalent functions, which is disallowed by the assumption, except where both functions are identically zero. Thus, the Lebesgue measure of the zero set of $\chi_i - \chi_j$ is zero (the borders of the $\mathbb{D}_{i,\alpha,\beta}$ also have measure 0). The proposition follows immediately. ■

A simple, but useful, corollary of this proposition is as follows.

*Corollary 1:* The segmentation of space into places via Eq. 15 is consistent, as per Definition 4, if $\chi_i$ and $\chi_j$ are real analytic functions on the domain $\mathcal{C}$, and $\chi_i \not\equiv \chi_j$ on $\mathcal{C}$.

If $a_i$ are piece-wise analytic functions and each applicability function $g_i$ is analytic on $\mathcal{A}$, then $\chi_i$ is piece-wise analytic on a partitioning of $\mathcal{C}$ (where the partitioning is a function of the domains on which $a_i$ are analytic).

The requirement that $\chi_i$, $\forall i$ are real analytic functions on all of $\mathcal{C}$ is sufficient but not necessary. In some cases this requirement is too restrictive; e.g. it prohibits binary (true/false) type features. The following result provides an useful augmentation.

*Proposition 2:* Suppose that $\chi_i = (\chi_i^d + \chi_i^a)\chi_i^b$ and $\chi_j = (\chi_j^d + \chi_j^a)\chi_j^b$, where $\chi_i^a$ and $\chi_j^a$ are real analytic functions on $\mathcal{C}$, and $\chi_i^d$ and $\chi_j^d$ are piecewise constant on $\mathcal{C}$. Moreover, $\chi_i^b$ and $\chi_j^b$ are functions taking values in $\{0, 1\}$ over all $\mathcal{C}$. Assume that $\chi_i^a - \chi_j^a \not\equiv C$ where $C$ is a constant. Then the segmentation of space into places is consistent, as per Definition 4.

*Proof:* Note first that with no loss of generality we can ignore the effect of $\chi_i^b$ and $\chi_j^b$ and consider only the remaining functions. $\chi_i^d$ is a constant function $\chi_i^d \equiv C_\alpha$ on each open domain $\mathbb{D}_{i,\alpha}$, where $\{\bigcup_\alpha \mathrm{cl}(\mathbb{D}_{i,\alpha})\} = \mathcal{C}$, and analogously for $\chi_j^d$. Then, $\chi_i - \chi_j$ is a piecewise real analytic function on each non-empty $\mathbb{D}_{ij,\alpha,\beta} \triangleq \mathbb{D}_{i,\alpha} \cap \mathbb{D}_{j,\beta}$, and $\chi_i - \chi_j \equiv \chi_i^a - \chi_j^a + C_\alpha - C_\beta$ on $\mathbb{D}_{ij,\alpha,\beta}$. The zero set of this function can only have a non-zero Lebesgue measure if $\chi_i^a - \chi_j^a$ is constant, which is disallowed. ■

The last proposition accounts for discrete-valued feature types to be used in admissibility functions as a special case (given that they are accompanied by a continuous component).

Features of the type $f_i(\mathbf{x}, t) : \mathcal{C} \to \{0, 1\}$ are useful since so-called visibility features are of this type. That is, a point $\mathbf{y}^* \in \mathcal{C}$ is either visible (1) or not visible (0) from another point $\mathbf{x} \in \mathcal{C}$. The support of a visibility feature $f_i(\mathbf{x}, t) : \mathcal{C} \to \{0, 1\}$ belongs to the class of so-called star-shaped sets; e.g. see [4].

In the final corollary, we show how two useful classes of feature functions can be combined in an applicability function to provide a consistent segmentation of space:

*Corollary 2:* Assume that

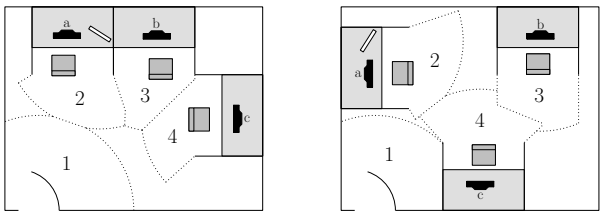
$$\chi_i = \Omega_i\left(\{a_k^b\}_{k \in M^b}\right)\left(\sum_{k \in M^d} \lambda_{ik} a_k^d + \chi_i^a\right) \quad (16)$$

where $a_k^b$ are binary-valued features from $A$, and $a_k^d$ are piece-wise constant functions taken from $A$. $\Omega$ is any logical expression on the $a_k^b$. Assume that $\chi_i^a - \chi_j^a \not\equiv C$ where $C$ is a constant. Then the segmentation of space into places is consistent, as per Definition 4.

### *A. Example 2 - Distance and Visibility Features*

As a theoretical illustration, consider a small office with three desks (see Figure 2(a)). The desks each have a computer screen and one additionally a framed picture. They are partially surrounded by partitions which block the view.

Four places have been assigned, all defined by different features ($t$ omitted for clarity):




(a) Configuration 1 (b) Configuration 2

Fig. 2. Two configurations of an office and their consequent place regions.

- $\mathcal{P}_1$ - "Close to door object"
  $\chi_1(\boldsymbol{x}) = f_{door_c}(\mathbf{x}) = \frac{\mathbf{1}}{\mathbf{1}+\|\boldsymbol{p}_{\mathbf{door}}-\boldsymbol{x}\|}$
- $\mathcal{P}_2$ - "Close to picture and picture visible"
  $\chi_2(\boldsymbol{x}) = f_{pic_v}(\boldsymbol{x}) \cdot f_{pic_c}(\boldsymbol{x}) = f_{pic_v}(\boldsymbol{x}) \cdot \frac{1}{1+\|\boldsymbol{p}_{pic}-\boldsymbol{x}\|}$
- $\mathcal{P}_3$ - "Close to computer b and in front of desk"
  $\chi_3(\boldsymbol{x}) = f_{desk_f}(\boldsymbol{x}) \cdot f_{comp2_c}(\boldsymbol{x})$
  $= f_{desk_f}(\boldsymbol{x}) \cdot \frac{1}{1+\|\boldsymbol{p}_{comp2}-\boldsymbol{x}\|}$
- $\mathcal{P}_4$ - "Close to computer c and computer c visible"
  $\chi_4(\boldsymbol{x}) = f_{comp3_v}(\boldsymbol{x}) \cdot f_{comp3_c}(\boldsymbol{x})$
  $= f_{comp3_v}(\boldsymbol{x}) \cdot \frac{1}{1+\|\boldsymbol{p}_{comp3}-\boldsymbol{x}\|}$

Here,

$$f_{pic_v}(\boldsymbol{x}) = \begin{cases} 1 & \text{if picture unoccluded from } \boldsymbol{x} \\ 0 & \text{otherwise} \end{cases}$$

and analogously for $f_{comp2_v}$ and $f_{comp3_v}$. The "in front of" feature is also binary:

$$f_{desk_f}(\boldsymbol{x}) = \begin{cases} 1 & \boldsymbol{x} \in \mathcal{X}_{desk} \\ 0 & \text{otherwise} \end{cases}$$

where $\mathcal{X}_{desk}$ is a region projecting straight outward from the edge of the desk – cf. Figure 2(b).

These applicability functions fulfill the requirements of Proposition 2, as the radial components have different centers. It is assumed that there is a threshold for the applicability functions, below which a point is not considered part of any of the places (hence the circular borders). In effect, the regions belonging to the four places "compete" for the space and the best match wins out at each point.

These features exemplify the different sorts of functional aspects that define places to a cognitive agent. In a real-world scenario, places would likely be characterized by a larger number of features combined, for increased robustness. For the same reason, the granularity of places would typically also be finer. Also, since the features would be selected autonomously by a robotic agent their definition might be less human-comprehensible than the above selection. Still, this discrepancy will ideally be kept small, so that the spatial conceptualization of human and robot are invariant to similar types of features.

In Figure 2(b), the same office is shown after a rearrangement of the desks. Note how the regions, though their shape and size have changed, remain well-defined and how the cognitively conceptualized places (in the sense of having functionally conceived features) maintain their semantic significance despite having entirely different metric properties.

## V. Navigation

The places discussed in Section III provide the segmentation of space into discrete units, and allow an agent to localize itself in the environment, by evaluating places' descriptor sets at its current location using its sensors. A map must, besides allowing for localization, provide a means for navigating through it. We do this in terms of *paths*, which represent the (potential) movement from one (start) place to another (goal) place. Just as places are defined by descriptors, so each path is associated with a *path precept*.

*Definition 5:* Let $\mathcal{S}$ represent the space of low-level sensor inputs available to the agent. Similarly, let $\mathcal{O}$ represent the space of low-level control outputs. Then, a path precept is a mapping from a low-level sensory state $s \in \mathcal{S}$ to a control output $o \in \mathcal{O}$:

$$\pi_i : \mathcal{S} \mapsto \mathcal{O} \tag{17}$$

A path is always associated with exactly one precept. $\mathcal{S}$ is of course given by the system instantiation, and may include virtual sensor modalities, such as local metric maps built over a period of time. It is in general a richer representation than the feature space $\mathcal{F}$, and allows for low-level considerations such as obstacle avoidance and other reactive behaviours.

The above definition is very general and admits path precepts that produce any sort of output. We therefore distinguish between *proper* and *improper* path precepts.

*Definition 6:* A *proper* path precept will, if applied continuously while moving from the start place of the path, bring the agent to the goal place.

Note that, in an unpredictable real-world application, this property of path precepts is a random variable; a precept might be more or less proper depending on its success rate. Also, a dynamic world implies that path precepts may cease to be proper due to changes in the environment.

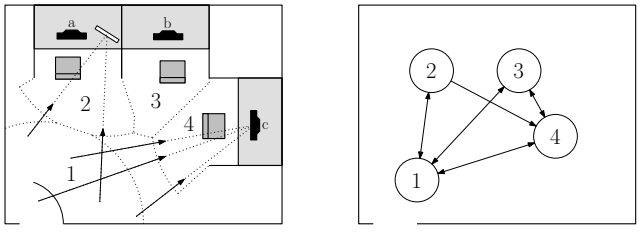
### A. Principles for path precepts

The fundamental attribute of a proper path precept is that the output brings the agent to the place to which the path is leading. Places, in turn, are defined in terms of descriptors. These two facts give rise to the following basic rule for creating proper path precepts:

*Remark 1:* A path precept should be defined such that it, given a sensory state, produces a control output that is expected to increase the relative (compared to those of competing places) applicability function of the goal place.

Thus, the form of the precept naturally arises from the descriptor that define places: A precept that keeps successfully increasing the applicability function must eventually reach the goal place; conversely, the goal cannot be reached without increasing it. Obviously, the method of accomplishing this can vary. Local hill-climbing approaches are general, but suffer from local maxima, whereas global maximization though more robust requires more information and sophisticated control. The actual control policy chosen will depend on available sensory information, control outputs, and efficiency considerations.

*Remark 2:* If the instantiation permits applicability to be evaluated outside of the immediate surroundings of the




(a) Paths leading from place 1 to places 2 and 4.

(b) Place graph for the office.

Fig. 3. Examples of paths.

current configuration $\boldsymbol{x} \in \mathcal{C}$ and if the control output is of an abstraction level that admits set-points in $\mathcal{C}$, then the following specialization of the above rule can be made:

$$\mathbf{x}^* = \underset{\mathbf{x}\in\mathcal{C}}{\operatorname{argmax}} \left( \chi_i(\mathbf{x}) - \max_{j\neq i} \chi_j(\mathbf{x}) \right) \tag{18}$$

where $\mathbf{x}^*$ is the set-point for the agent's controller, $i$ is the goal place, and $\chi_k$ the applicability function for place $k$.

The above principles may still leave some ambiguity as to the precise contents of the precept; different descriptors may suggest entirely different movement rules, and the way different descriptors change with movement may be more or less easy to predict in varying sensory circumstances. Any implementation that mixes different types of descriptors will therefore require a facility for estimating the applicability of the goal place at a distance – or at least, caching such information when it is available – and, based on this, producing a local navigation goal for lower-level navigation to carry out.

Apart from being proper, a path precept also needs to be well-defined for all sensor states. Moreover, it should be efficient in execution (i.e. minimizing the time, distance, energy etc. necessary to reach the goal) and efficient to evaluate (i.e. computationally).

### B. Example

As a simple example of path precepts derived from place descriptors, regard the office in Figure 2(a). The simplicity of each place's applicability function makes it easy to define path precepts through Remark 2. Take for example $i = 2$:

$$\mathbf{x}^* = \underset{\mathbf{x}\in\mathbb{R}^2}{\operatorname{argmax}} \left( \chi_2(\mathbf{x}) - \max_{j\neq 2} \chi_j(\mathbf{x}) \right) = p_{pic}$$

In other words, the precept is simply to move towards the picture in order to reach place $\mathcal{P}_2$. Figure 3(a) illustrates how different points in place $\mathcal{P}_1$ will give rise to different trajectories into place $\mathcal{P}_2$, and correspondingly for place $\mathcal{P}_4$. Note that once the agent enters the goal place and detects this, there's no point in continuing to the set point; the path precept is simply meant to take it within the boundary of the place.

The above path precept for $\mathcal{P}_2$ does not necessarily work as well in $\mathcal{P}_3$ and $\mathcal{P}_4$, however. If it is assumed that the agent is unable to detect the picture behind the partition (such as by virtual sensing), or if it lacks the obstacle avoidance capacity to approach $p_{pic}$ except by a straight line, then this path precept is not proper to the paths from $\mathcal{P}_3$ and $\mathcal{P}_4$ to $\mathcal{P}_2$.

In the same way, the natural path precept from $\mathcal{P}_2$ to $\mathcal{P}_3$ (moving toward computer b) is not proper to that path. Figure 3(b) shows a graph containing the paths which have proper precepts. Note that the path from $\mathcal{P}_2$ to $\mathcal{P}_4$ is more proper than its reverse.

The distinction between proper and improper path precepts is not clear-cut even in this simple example: there are points in $\mathcal{P}_3$ from which the picture in $\mathcal{P}_2$ is visible, and points in $\mathcal{P}_2$ where the computer in $\mathcal{P}_4$ cannot be seen.

If the room is rearranged, as in Figure 2(b), then while the path precepts remain the same (being defined as in Remark 2) they will no longer be proper or improper to the degree indicated by the graph in Figure 3(b). An agent relying on that information to navigate in the office may fail to do so, but can update its representation by invalidating paths that fail and creating new ones from the unchanged precepts.

## VI. Localization

According to the definition of *places* in Section III, given the true values of place descriptors (features and spatial relations) $D_t = (\zeta_{1,t}, \ldots, \zeta_{n_f,t}, \rho_{1,t}, \ldots, \rho_{n_r,t})$ obtained at time $t$ for location $\boldsymbol{x}(t)$, the place to which that $\boldsymbol{x}$ corresponds is uniquely identified. Consider a function $D(\boldsymbol{x}, t) = (f_1(\boldsymbol{x}, t), \ldots, f_{n_f}(\boldsymbol{x}, t), r_1(\boldsymbol{x}, t), \ldots, r_{n_r}(\boldsymbol{x}, t))$ that provides the true values of place descriptors for location $\boldsymbol{x}$ and time $t$. Then, for $D_t = D(\boldsymbol{x}(t), t)$, the true place is given by $L_t \triangleq i : D_t \in \mathcal{P}_i$.

However, in the real world an agent is moving through space, following paths to get from place to place and needs to maintain its localization in the face of uncertainty. Let us denote the observation of all descriptors at time $t$ as $\hat{D}_t = D_t + e_t$, where $e$ is an error. We view the agent's progress from place to place as a Markov process with $L_t$ the state at (discrete) time $t$ and $\hat{D}_t$ the measurement. Localization is then carried out iteratively according to the following formula:

$$\begin{aligned} & p\left(L_t \mid \{\hat{D}\}_t, \{\alpha\}_{t-1}\right) \\ &= \sum_{L_{t-1}} p\left(L_t \mid L_{t-1}, \hat{D}_t, \alpha_{t-1}\right) \\ &\quad \times p\left(L_{t-1} \mid \{\hat{D}\}_{t-1}, \{\alpha\}_{t-2}\right) \end{aligned} \tag{19}$$

where $\{\hat{D}\}_t$ represents all measurements up until time $t$, and equivalently for the actions $\alpha$.

The probability update in Eq. 19 is computed as follows:

$$\begin{aligned} & p\left(L_t \mid L_{t-1}, \hat{D}_t, \alpha_{t-1}\right) \\ &= \gamma \cdot p\left(\hat{D}_t \mid L_t\right) p\left(L_t \mid L_{t-1}, \alpha_{t-1}\right) \end{aligned} \tag{20}$$

Here, $\gamma$ is a normalization constant, and $\alpha_t$ is the action taken at time $t$; that is, a choice of a path to follow and an according path precept.

The factors in Eq. 20 represent respectively the measurement integration step, and the prediction step, of the localization update.

### A. Prediction

The prediction step encapsulates the probability of transitioning from one place to another given the action $\alpha_t$. If $\boldsymbol{x}_t$ and $\boldsymbol{x}_{t+1}$ are the configurations at time $t$ and $t+1$ respectively, then

$$\begin{aligned} & p\left(L_{t+1} \,|\, L_t, \alpha_t\right) \\ &= \int\limits_{\boldsymbol{x}_{t+1}} p(L_{t+1} \,|\, \boldsymbol{x}_{t+1}) p(\boldsymbol{x}_{t+t} \,|\, L_t, \alpha_t) d\boldsymbol{x}_{t+1} \\ &= \iint\limits_{\substack{\boldsymbol{x}_{t+1} \in \mathcal{C}_{L_{t+1}} \\ \boldsymbol{x}_t}} 1 \cdot p(\boldsymbol{x}_{t+1} \,|\, \boldsymbol{x}_t, \alpha_t) p(\boldsymbol{x}_t \,|\, L_t) d\boldsymbol{x}_t d\boldsymbol{x}_{t+1} \end{aligned} \tag{21}$$

The factor $p(\boldsymbol{x}_{t+1} \mid \boldsymbol{x}_t, \alpha_t)$ represents the evolution of the exact configuration during the transition, and can be computed via the Fokker-Planck equation (see e.g. [11]); we assume the continuous-time process can be written:

$$\begin{aligned} d\boldsymbol{\xi} &= f_\alpha(\boldsymbol{\xi}) d\tau + N(\boldsymbol{\xi}) d\eta \\ \boldsymbol{\xi}(0) &= \boldsymbol{x}_t \\ \boldsymbol{x}_{t+1} &= \boldsymbol{\xi}\left(\min\{\tau : S_\alpha\left(\boldsymbol{\xi}(\tau), \tau\right) = 0\}\right) \end{aligned} \tag{22}$$

where $f_\alpha$ represents the motion model, given the chosen path precept, and $d\eta$ represents the random evolution of a stochastic process such as a Brownian motion. $N$ is a configuration-dependent transformation of the process noise. The transition ends when the stopping condition $S$, given by the path precept, evaluates to $0$.

The resulting integral is very difficult to compute in general, and an analytic solution will not be feasible except for the very simplest cases.

Because of this, it may be more profitable to view the state transition probabilities as hidden model parameters:

$$p\left(L_{t+1} = j \mid L_t = i, \alpha\right) = \theta_{i,j,\alpha} \tag{23}$$

Given an initial estimate for $\theta_{i,j,\alpha}$ and observations of outcomes of action execution in a real or simulated setting, the parameters can be iteratively estimated through Expectation-Maximization.

The basic constraint is that $\sum_i \theta_{i,j,\alpha} = 1$. Reasonable initial estimates will vary with instantiation, and may be taken from appropriately defined relations; as an example, a transition to a nearby or adjacent place might be assigned a higher probability by default. The simplest assumption is that of uniform probability: $\theta_{i,j,\alpha} = 1/n_P$ where $n_P$ is the number of places.

### B. Measurement integration

After the action is finished, the measurement step incorporates observations of descriptors into the probability distribution. As is seen below, this expression is complicated by the fact that knowing the place does not imply a probability distribution over exact locations $\boldsymbol{x}$, nor over descriptor values $D$.

Observed descriptor values are conditionally independent of place, given true descriptor values $D'$:

$$\begin{aligned} & p\left(\hat{D}_t \mid L_t\right) \\ &= \int_{D' \in \mathcal{D}} p\left(\hat{D}_t \mid D'\right) p\left(D' \mid L_t\right) dD' \end{aligned} \tag{24}$$

The first factor is simply the likelihood of the observation. Expressed using the probability distribution of the measurement error, it becomes:

$$p\left(\hat{D}_t \mid D'\right) = p_e\left(\hat{D}_t - D'\right) \tag{25}$$

If observation errors are taken to be conditionally independent, given the true descriptor values, the likelihood function can be written:

$$\begin{aligned} & p\left(\hat{D}_t \mid D'\right) \\ &= \prod_{i=1}^{n_f} p\left(\hat{\zeta}_{i,t} \mid \zeta_i\right) \prod_{i=1}^{n_r} p\left(\hat{\rho}_{i,t} \mid \rho_i\right) \\ &= \prod_{i=1}^{n_f} p_{e_i}\left(\hat{\zeta}_{i,t} - \zeta_i\right) \prod_{i=1}^{n_r} p_{e'_i}\left(\hat{\rho}_{i,t} - \rho_i\right) \end{aligned} \tag{26}$$

where $e_i$ and $e'_i$ are the errors associated with the measurement of feature $i$ and relation $i$, respectively.

The second factor in Eq. 24 represents the way descriptor values are distributed inside places. One way of dealing with it is to assume a normalized distribution of $D'$ over $\mathcal{P}_i$, i.e. a constant. However, this distribution is dependent on the details of the instantiation. If it cannot be modeled or estimated, another approach is to evaluate

$$\begin{aligned} & p\left(D' \mid L_t\right) \\ &= \int_{\boldsymbol{x} \in \mathcal{C}} \delta\left(D' - D\left(\boldsymbol{x}, t\right)\right) p\left(\boldsymbol{x} \mid L_t\right) d^m \boldsymbol{x} \\ &= \int_{\boldsymbol{x} \in \Psi} \frac{p\left(\boldsymbol{x} \mid L_t\right)}{|\nabla D(\boldsymbol{x}, t)|} d^{m-1} \boldsymbol{x} \end{aligned} \tag{27}$$

where $\Psi$ denotes all $\boldsymbol{x}$ which satisfy $D' = D(\boldsymbol{x}, t)$. $\delta$ is the Dirac distribution, and the final step uses the generalized scaling property of integrals over Dirac distributions. $m$ is the dimension of $\mathcal{C}$.

$p\left(\boldsymbol{x} \mid L\right)$ can be modeled either as a constant over $\mathcal{C}_{L_t}$ or estimated based on observations. If a place is defined in terms of an applicability function, the spatial information encoded in it can also be used to model this distribution.

## VII. Discussion

Despite the fact that the framework presented in the previous section represents a certain view on the structure of a cognitive map, it is also very general and allows for expressing many existing approaches as specific cases. Consider for instance the topological map constituting a part of the Multi-Layered Conceptual Spatial Representation presented in [14]. The authors propose to create a topological representation on top of a two-dimensional metric line map,

and ground each topological node around a point anchored to the metric map. Such approach can be easily expressed in our framework if we define a feature $\zeta = f(\boldsymbol{x}, t) = \boldsymbol{x}$, where $\boldsymbol{x} \in \mathcal{C} = \mathbb{R}^2$ represents the coordinates on the metric map, and a set of applicability functions $\{g_i(\zeta)\}_{i=1}^{n_t}$ such that $g_i(\zeta) = 1/(1 + |\boldsymbol{t}_i - \zeta|)$ for each of the $n_t$ topological nodes, where $\boldsymbol{t}_i$ is the center of the node expressed in the coordinates of the metric line map.

The generality of the presented approach can accommodate a very wide range of different methods for abstracting space into places. Exact grid decomposition [1] as well as fixed decomposition can both be described in terms of this framework, given properly chosen features, as can the "islands of reliability" of [12]. Even a system such as the Spatial Semantic Hierarchy [6] is possible to express in these terms; however, to accomplish this, a relatively high level of abstraction must be assumed for the features and the sensor input. Nevertheless, it is our expectation that such requirements will not apply in general to powerful and cognitively well-founded instantiations of this framework.

### A. Future work

Possible directions in which to extend this work include:

*1) Feature selection:* Within this paper we have assumed a set of features as given. In a practical system, an agent will have access to high-dimensional low-level sensor data and the features used for building scenes will need to be abstracted from this data. This can be done in either a pre-programmed or an automatic manner.

*2) Virtualized sensors:* Herein, features are defined as functions of single points in configuration space; in effect, a feature is conceived of as an abstract sensor output while the agent is at that point. In practice, techniques that allow information to be integrated over time may serve as "virtual" sensor input permitting more advanced features to be defined.

*3) Clustering:* This paper has suggested one way of clustering the feature space into scenes using applicability functions. Methods for automatic and dynamically updated clustering could be applied.

*4) Spatial reasoning:* One principal use for segmenting space, in a cognitive systems context, is high-level spatial reasoning, planning, learning and communication. It would be useful to explore the implications of a feature-based place concept when integrated as a component of a full cognitive system.

## VIII. Conclusions

We have presented a general framework for building a spatial map based on places and scenes which supports localization and navigation using arbitrary features and higher-level spatial relations. We suggested how the framework would be used to instantiate a system with cognitively plausible features, as well as how to extract precepts for moving from one place to another. Probabilistic expressions used for localization in the framework were presented and the necessity for additional assumptions was highlighted.

The framework has been shown to entail existing spatial representations. In the future, we hope to demonstrate instantiations built directly on the proposed framework, which will prove the viability of the approach and its usefulness in higher-level reasoning.

## References


[1] J. Barraquand and J.-C. Latombe. Robot motion planning: a distributed representation approach. *International Journal of Robotics Research*, 10(6), 1991.

[2] H. Bay, T. Tuytelaars, and L. Van Gool. SURF: Speeded up robust features. In *Proceedings of the 9th European Conference on Computer Vision (ECCV)*, Graz, Austria, 2006.

[3] M. Cummins and P. Newman. FAB-MAP: Probabilistic localization and mapping in the space of appearance. *International Journal of Robotics Research*, 27(6), 2008.

[4] A. Ganguli, J. Cortes, and F. Bullo. Maximizing visibility in non-convex polygons: Nonsmooth analysis and gradient algorithm design. *SIAM Journal on Control and Optimization*, 45(5), 2006.

[5] A. S. Huang and S. Teller. Non-metrical navigation through visual path control. Technical Report MIT-CSAIL-TR-2008-032, 2008.

[6] B. Kuipers, J. Modayil, P. Beeson, M. MacMahon, and F. Savelli. Local metrical and global topological maps in the hybrid spatial semantic hierarchy. In *Proceedings of the International Conference on Robotics and Automation (ICRA)*, New Orleans, USA, 2004.

[7] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2), 2004.

[8] M. J. Milford and G. Wyeth. Mapping a suburb with a single camera using a biologically inspired slam system. *IEEE Transactions on Robotics, Special Issue on Visual SLAM*, 24(5), 2008.

[9] A. Pronobis, O. Martinez Mozos, and B. Caputo. SVM-based discriminative accumulation scheme for place recognition. In *Proceedings of the International Conference on Robotics and Automation (ICRA)*, Pasadena, CA, USA, 2008.

[10] D. A. Randell, Zhan Cui, and A. G. Cohn. A spatial logic based on regions and connection. In *Proceedings of the International Conference on Knowledge Representation and Reasoning*, 1992.

[11] H. Risken and T. K. Caugheyz. The fokker-planck equation: Methods of solution and application, 2nd ed. *Journal of Applied Mechanics*, 58(3):860–860, 1991.

[12] S. Simhon and G. Dudek. A global topological map formed by local metric maps. In *Proceedings of the International Conference on Intelligent Robots and Systems (IROS)*, 1998.

[13] A. Torralba, K. P. Murphy, W. T. Freeman, and M. A. Rubin. Context-based vision system for place and object recognition. In *Proceedings of the International Conference on Computer Vision (ICCV)*, Nice, France, 2003.

[14] H. Zender, Ó. Martínez Mozos, P. Jensfelt, G.-J. M. Kruijff, and W. Burgard. Conceptual spatial representations for indoor mobile robots. *Robotics and Autonomous Systems*, 56(6), June 2008.

# Multi-Layered Conceptual Spatial Mapping

Hendrik Zender
zender@dfki.de

Technical Report, 2010

**Abstract**

In this paper, we identify structuring of space and categorization of large-scale space as two important aspects of spatial understanding for embodied cognitive systems. In order to enable an autonomous agent to engage in a situated dialogue about its environment, it needs to have a human-compatible spatial understanding, whereas autonomous behavior, such as navigation, requires the agent to have access to low-level spatial representations. Addressing these two challenges, we present an approach to multi-layered conceptual spatial mapping. We embed our work in a discussion of relevant research in human spatial cognition and mobile robot mapping.

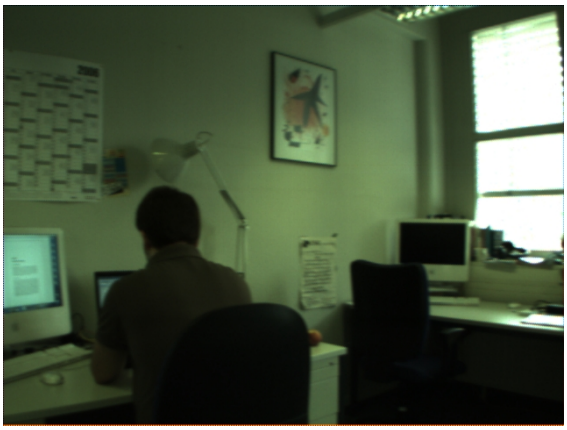
## 1 Motivation and Background

We are driven by the research question of *spatial understanding* and its connection to acting and interacting in indoor environments. We want to endow autonomous embodied agents with the capability to conduct spatially *situated dialogues*. For this the agent must be able to understand space in terms of concepts that can be expressed in, and resolved from natural language.

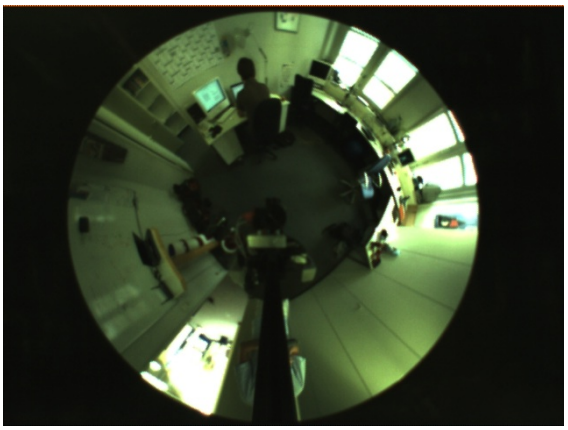
We start from the assumption that the environment is not instrumented in order to facilitate the mapping problem. The kinds of environments that we are interested in are indoor spaces that are designed by humans for humans – and that are intuitively and easily *understood* by humans. This includes ordinary and everyday indoor office environments or apartments that are populated by humans working and living there. This also includes virtual spaces that are designed in such a way that humans who control an avatar using a 3D client software perceive of them as if they were realistic models of natural physical spaces. We call this class of environments that are made and designed by humans for being used and populated by humans *human-oriented environments*. Figure 2 demonstrates examples of different human-oriented environments in which autonomous agents have to operate. Figure 1 shows how a robot's sensors (cameras and laser range finders) perceive such an environment.[1]

There exist many different approaches for equipping autonomous embodied agents, most notably mobile robots, with spatial models. The problem is that these models are usually specifically tailored for the tasks the agent is supposed to fulfill. This means that the features of the spatial representation are typically only *meaningful* with respect to the algorithms that work on these representations. These include, for instance, occupancy grid maps (see Figure 3a on page 4 for an

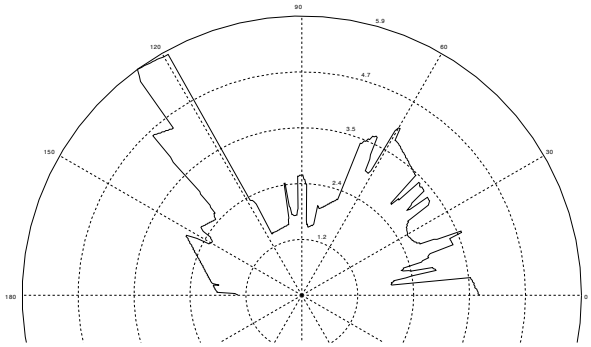
[1] Still images and sensor readings taken from the CoSy Localization Database (COLD) [32].



(a) Perspective image taken from a digital camera mounted on the top platform of the robot, facing forward (height: 140cm, field of view: 68.9°).



(b) Omnidirectional image taken from a digital camera facing up towards a hyperbolic mirror (height: 116cm, field of view: 360°).




(c) Frontier of the corresponding laser range scan taken at a vertical height of 30cm in parallel to the floor plane (field of view: 180°).


(d) The mobile robot used for acquiring the data. The cameras and the laser scanner can be seen on the top and bottom platforms, respectively.

Figure 1: An office environment "seen" from the point of view of a robot using different sensors.

(a) Autonomous mobile robots – left: Explorer and, right: Dora – operating in an office building.

(b) A virtual character in a household environment within the Twinity world.

Figure 2: Examples of human-oriented environments.

example[2]), which address the challenge of representing which parts of an environment are likely to be free and unobstructed, and which ones contain potential obstacles [39], or line maps that represent static features of the environment for the purpose of *simultaneous localization and mapping (SLAM)*, illustrated in Figure 3b[3].

In contrast to this, what we need are human-like features. In order to be able to talk in and about space, the agent needs to abstract from its internal, machine-compatible representations of space to a level that is at least comparable to the way humans perceive of space.

(a) Underwater grid map of a marina in San Pere Pescador, Spain [22].

(b) Line-feature map of an indoor environment.

Figure 3: Examples of robotic spatial representations for SLAM.

*Spatial understanding* comprises two aspects. For one, it concerns *structuring* of spatial organization. That is, which are the units a human-oriented environment is composed of? Secondly, it concerns *categorization* of space. That is, which are the concepts that describe these spatial units, and how are they determined? We call spatial knowledge representations that address these issues *human-compatible representations* of space.

To this end the work presented in this thesis builds upon and extends the author's previous research on *multi-layered conceptual spatial mapping* [44, 45] in the tradition of approaches like the *(Hybrid) Spatial Semantic Hierarchy* [25, 26, 1], the *Route Graph* model [42, 23], *hybrid maps* [4], and *multi-hierarchical semantic maps* for mobile robots [16, 15]. The approach is inspired by human cognition. On the lower layers it contains sensor-based representations. These are abstracted into basic categories (free space vs. occupied space, areas vs. humans vs. objects, rooms vs. corridors, etc.). The basic spatial relation is spatial containment, corresponding to the container schema, which is among the most prominent, most important, and most fundamental schemata in human cognition [27].

[2]Image generated from the marina dataset [35], courtesy of Shanker Keshavdas [22].
[3]Image taken from [47].

## 1.1 Structuring space

Research in cognitive psychology addresses the inherently *qualitative* nature of human spatial knowledge. It tries to answer the question how the human mind represents spatial information in a so-called *cognitive map*. Following the results of empirical studies, it is nowadays generally assumed that humans adopt a *partially hierarchical* representation of spatial organization [38, 29]. The basic units of such a qualitative spatial representation are *topological* regions [8], which correspond to more or less clearly bounded spatial areas. The borders may be defined *physically*, *perceptually*, or may be purely *subjective* to the human. It has been shown that even in natural environments without any clear physical or perceptual boundaries, humans decompose space into topological hierarchies by clustering salient landmarks [19]. In our approach, topological areas are the primitive units of the conceptual map that is used for human-robot interaction and dialogue, and the basic spatial relation is topological inclusion.

Recent advances in cognitive neuroscience have found evidence for brain structures that supply the topological representations of the so-called "place-cells" with a metric one encoded in the so-called "grid cells" [20]. This does not contradict the assumption that the global-scale representation of *large-scale space* in the cognitive map is a topological one. It rather provides insight into how local scenes, i.e., *small-scale space*, might be represented in the human mind and speaks in favor of a multi-layered, hybrid representation of space in the cognitive map.

### 1.1.1 Large-scale space and small-scale space

There is an important distinction to make when investigating any kind of spatially situated behavior, be it acting, planning, observing, learning, or communicating, namely if it pertains to space that constitutes the agent's immediate surroundings, or if it pertains to larger spatial structures. The dichotomy between *small-scale space* and *large-scale space* for human spatial cognition [18, 17] is central to the work presented in this thesis.

[24] defines large-scale space as "a space which cannot be perceived at once; its global structure must be derived from local observations over time," whereas small-scale space consist of the here-and-now. For example, a drawing is a large-scale space "when viewed through a small movable hole, while a city can be small-scale when viewed from an airplane" [24]. In more common everyday situations, an office environment, one's house, a city, or a university campus are large-scale spaces. A table-top or a particular corner of one's office are examples of small-scale space.

This crucial distinction is reflected in the spatial models (WP3) developed in the CogX project as well as the methods for situated natural language processing (WP6).

### 1.1.2 Segmenting and partitioning space

As mentioned earlier, it is important that autonomous agents which are supposed to interact with humans in a human-oriented environment have a notion of spatial units that are also meaningful for humans. Topological regions are such units that are meaningful to humans. We call the units of indoor spaces *areas*. We distinguish between two basic kinds of areas. *Rooms* are spatial areas whose primary purpose is defined by the kinds of actions they afford. The other major class of indoor areas are *passages* whose primary purpose is to link rooms and provide access to other spatial areas.

The challenge for intelligent agents is to autonomously build spatial representations that are composed of such areas. The previously mentioned distinction between physical, perceptual and

subjective boundaries of topological areas corresponds to a *spatial segmentation* along geometric features versus functional features. In indoor environments, walls are the physical boundaries of areas. They determine the geometric layout of the space they surround. Functional features, can be determined by specific objects – but also by the spatial layout and the composition of the objects and their surroundings.[4] Similarly, the gateways that link areas can be defined geometrically or on a functional-perceptional basis.

However, as we showed in the previous sections, the sensors of a robot are not particularly geared towards perceiving architectural structures. Neither do computer vision methods exist that allow to visually recognize arbitrary objects – let alone their functional affordances. Currently, the main purpose of robotic exteroceptive sensors is to discriminate free space from physical obstacles, and to provide a means for localizing the robot with respect to local landmarks. It is therefore necessary to make use of other cues to *segment* an environment into topological units.

A special kind of free space are geometrically bounded *gateways*. In a spatial representation that is based upon free space and its inter-connectivity, gateways play an important role in structuring and segmenting free space. In a map that only implicitly represents the boundaries of spatial areas, gateways divide space into regions that belong to one spatial area from regions that belong to other spatial areas. "Cognitively this allows the world to be broken up into smaller pieces" [5]. Gateways constitute an important factor for spatial cognition and navigation of autonomous agents in large-scale space [6]. [7] explains the special role of gateways for autonomous robots like this:

> *"In buildings, these [gateways] are typically doorways... Therefore, a gateway occurs where there is at least a partial visual separation between two neighboring areas and the gateway itself is a visual opening to a previously obscured area. At such a [location], one has the option of entering the new area or staying in the previous area."*

Likewise, our approach is based on the assumption of the importance of gateways (especially doorways) for human-compatible spatial representations of human-oriented environments. Later we show how our approach makes use of information about doorways in order to maintain a representation that is composed of rooms and other spatial areas (e.g., corridors).

### 1.1.3 Hierarchical subdivision of space

One prominent spatial relation we experience physically and abstractly every day is spatial *containment*. [11] consider the space within a room as a small-scale space in which people experience cognitive image schemata, e.g., the *container-surface schema*. However, people routinely employ the same schemata to larger structures, for example when saying "the bench is in the garden" [27]. Similar to objects that are *inside* a room, streets are *in* a city, and several districts form a country. The space around us can thus be decomposed into smaller units, or can combine with other spatial units to larger regions. The container schema can – with a few constraints – also be applied to large-scale space – at least when considering objects of comparable size and similar observation scale [36].

Containment of objects or spatial units is a productive schema for spatial language [9], and one of the structuring principles in the cognitive map [38, 29]. Likewise, hierarchical subdivisions of space are a basic topological relation for *geographical information systems* (GIS) [28, 40].

[4] Strictly speaking, the presence of a coffee machine alone does not turn a room into a kitchen – it could as well be a storeroom. The space in the room must afford the preparation of coffee, just as the coffee machine must be reachable and usable.

Topological hierarchies can be expressed as spatial-relation algebras, which, unlike usual computational geometry-based calculations, "rely on symbolic computations over small sets of relations. This method is very versatile since no detailed information about the geometry of he objects, such as coordinates of boundary points or shape parameters, is necessary to make inferences" [11]. This makes them a prime candidate for a basic human-compatible relation to structure and subdivide space.

Conceptually, containment does not form a strict hierarchy. One spatial region can be contained in several different spatial regions, which, in turn, might not be in a containment relation. Consider, for example, an intersection of two corridors. While the intersection itself forms a spatial region, it can also be assumed to be a part of each individual corridor. The representation of spatial abstraction hierarchies is thus rather a *partially ordered set* (poset) [21].

**Definition 1** (Partially ordered sets (posets) [21])**.**

Let $P$ be a set. A *partial order* on $P$ is a binary relation $\leq$ on $P$ such that, for every $x, y, z \in P$:

1. $x \leq x$ (reflexive)
2. if $x \leq y$ and $y \leq x$, then $x = y$ (antisymmetric)
3. if $x \leq y$ and $y \leq y$, then $x \leq z$ (transitive)

A set $P$ with a reflexive, antisymmetric and transitive relation (*order relation*) $\leq$ is called a *partially ordered set* (or *poset*). For every partially ordered set $P$ we can find a new poset, the *dual* of $P$, by defining that $x \geq y$ is in the dual if $y \leq x \in P$. Any statement about a partially ordered set can be turned into a statement of its dual be replacing $\leq$ with $\geq$, and vice versa. $\geq$ is called the *inverse* of $\leq$.

In DR6.2, we show how a hierarchical subdivision of space provides the basic structure for the production and understanding of spatially situated language.

## 1.2 Categorizing space

Aside from the functionality of the cognitive map, another relevant question from cognitive science is how people categorize spatial structures. Categories determine how people can interact with, and linguistically refer to entities in the world. *Basic-level categories* represent the most appropriate name for a thing or an abstract concept. The basic-level category of a referent is assumed to provide enough information to establish equivalence with other members of the class, while distinguishing it from non-members [3, 37]. We draw from these notions when categorizing the spatial areas in the robot's *conceptual map*. We are specifically concerned with determining appropriate properties that allow a robot to both successfully refer to spatial entities in a situated dialogue between the robot and its user, and meaningfully act in its environment.

Our work rests on the assumption that the basic-level categories of spatial entities in an environment are determined by the actions they afford. Many types of rooms are designed in a way that their structure and spatial layout afford specific actions, such as corridors, or staircases. Other types of rooms afford more complex actions. These are in most cases provided by objects that are located there. For instance, the concept 'living room' applies to rooms that are suited for resting. Having a rest, in turn, can be afforded by certain objects, such as couches or TV sets. We thus conclude that besides basic geometric properties, such as shape and layout, the objects that are located in a room are a reliable basis for appropriately categorizing that room.

# 2 Representing Space at Different Levels of Abstraction

If an autonomous agent is required to perform navigation tasks, it must have access to low-level spatial representations that are suitable for fine-grained hardware control. These are typically *quantitative* spatial representations, such as *metric coordinate systems*. Metric maps rely on accurately measurable distances and dimensions. The sensors modern robots are typically equipped with, such as time-of-flight cameras or laser range finders, provide quite exact measurements of free and occupied space in the robot's surrounding. Such sensor readings are hence often stored in metric maps of different kinds. Metric maps are also an obvious choice for online avatars because they can have easy access to the virtual world, which typically consists of 3D models.

Humans, on the other hand, use the topological structuring of space to form a more *qualitative* sense of space. This is reflected in natural language, which is full of vague, qualitative spatial expressions. In order to be able to communicate successfully and naturally with humans, autonomous conversational agents must be able to establish such a quantitative spatial understanding on the basis of the low-level maps they can build from their sensory input.

To this end, we present *multi-layered conceptual spatial mapping*. The approach addresses the problems of human-compatible structuring and categorization of space. It comprises spatial representations at different levels of abstraction, ranging from low-level metric maps to symbolic conceptual representations.

Figure 4 on the following page shows two instantiations of the multi-layered conceptual spatial mapping principle. The spatial representation in Figure 4a on the next page is the basis for the integrated robotic systems of the CogX project (WP7). More recently, [34] presented a refined approach to multi-layered mapping, in which, most notably, the representations of the lower map layers were re-defined. The integrated robotic system Dora [43] makes use of this refined multi-layered map. It is illustrated in Figure 4b on the following page.

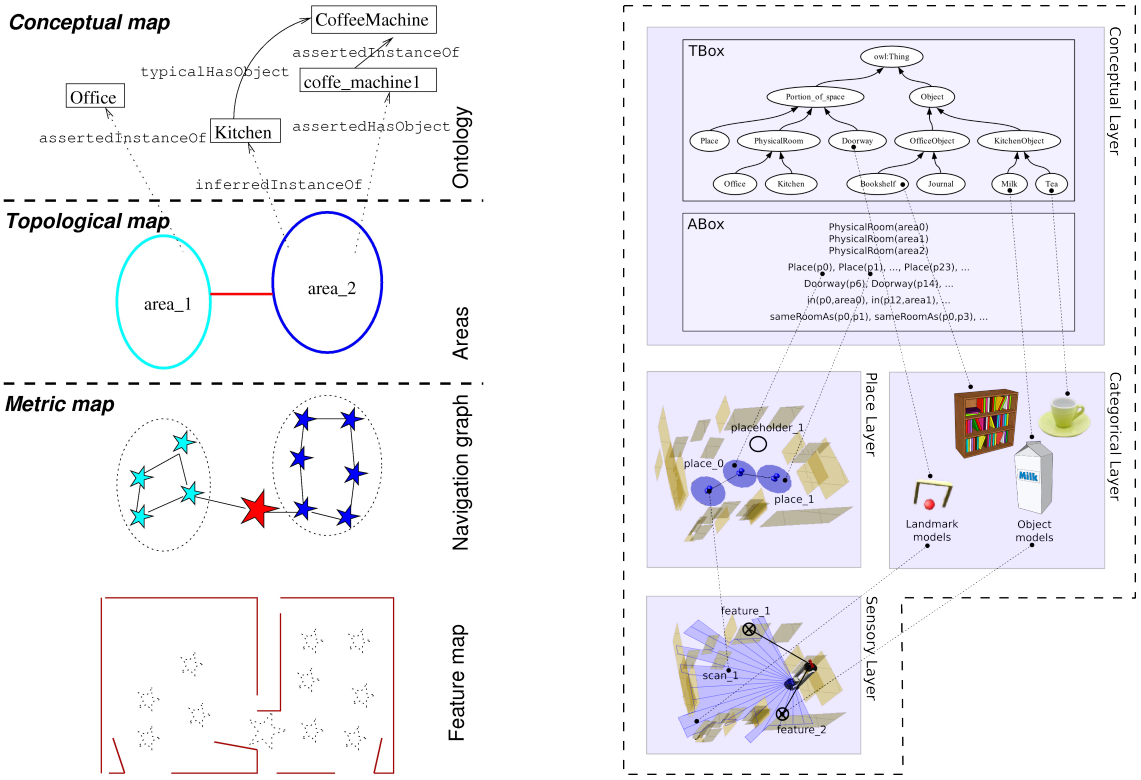
In the following sections we outline the different spatial representations underlying the individual abstraction layers.

## 2.1 Related work

Recently, a number of methods originating in robotics research have been presented that construct multi-layered environment models. These layers range from metric sensor-based maps to abstract conceptual maps that take into account information about objects acquired through computer vision methods. [41] suggest a hierarchical probabilistic representation of space based on objects. The work by [16, 15] presents an approach containing two parallel hierarchies, spatial and conceptual, connected through anchoring. Inference about places is based on objects found in them. This approach is based on the Multi-AH-graph model by [13]. The work by [10] creates a metric map through a guided tour. The map is then segmented into discrete rooms according to the labels given by the instructor. Furthermore, the *Hybrid Spatial Semantic Hierarchy* (HSSH), introduced by [1], allows a mobile robot to describe the world using different representations, each with its own ontology.

## 2.2 The different map layers

In the following, we briefly describe the properties of the individual layers. The conceptual map layer is central to the work presented in this thesis. The other layers, i.e., the metric, navigation, and topological layers will be referred to as the "lower layers" of the spatial model. They are outside




(a) Illustration of a multi-layered conceptual spatial map.

(b) COARSE (Cognitive lAyered Representation of Spatial knowledgE) [34]. Adapted from [43].

Figure 4: Two instantiations of the multi-layered mapping principle.

the scope of this thesis. While they are important for robot navigation and self-localization, their sole relevance to the work of this thesis is that they provide input to the conceptual layer based on perception of the real world.

Unless the agent is equipped with a form of external localization – such as robots acting in instrumented environments (which, in turn, are faced with their own challenges [12]), or avatars that operate in the 3D coordinate system of the virtual world – it must be equipped with sensors that allow it to perceive its surroundings. In the simplest case, such sensors are only used to prevent the robot from hitting an obstacle[5] or to enable the robot to move to a fixed target position.[6] This, however, does not amount to much spatial understanding other than a robot-centric frame of reference that captures the here-and-now small-scale space.

An understanding of large-scale space requires that the agent at least be able to represent – i.e., remember and retrieve – landmarks that are outside the currently observable part of space. Some approaches to mapping of large-scale space generate metric maps, ranging from interconnected patches of local maps [2] to larger, global metric maps of the whole operating environment [14]. In contrast, there are other approaches to mapping of large-scale space that do without local metric maps, but rather represent the positions of landmarks with respect to each other in terms of *control laws* that take the robot from one landmark to another [25].

Such maps, referred to as either *metric maps* or called the *sensory map layer* serve the principal purpose of allowing the robot to safely navigate its environment while staying localized within its representation of large-scale space. This self-localization can be performed in an absolute frame of reference or in a relative frame of reference with respect to a local landmark. As a result, such maps are essentially representations of *free and reachable* space and its *connectivity*, rather than faithful models of the architectural structure around that free space.

In order to allow for efficient path planning it is common practice to abstract away from sensor-based metric maps. The first abstraction step is *discretization* of the continuous metric space. Examples of such a discretization are *free-space markers* [31] which are used to form a *navigation graph map layer* in the implementation in the CoSy Explorer [47]. Recently [33] introduced the notion of *places* to form an intermediate map layer, which is part of the Dora integrated robotic system.

This level of discretization provides a basic notion of the topological structure of an environment. However, the discrete units are not guaranteed to be meaningful to humans. It is thus necessary to aggregate the units of the intermediate layer into *human-compatible spatial units*, such as rooms.

This then provides a *topological partitioning* that can be used for *human-compatible structuring* and *categorization* of space. In this view, the exact shape and boundaries of an area are irrelevant. Basic notions that are represented in such a map are *adjacency* and *connectivity*.

Together, the intermediate discretization layer and the topological layer provide a *symbolic abstraction* over continuous, sensor-based metric data. The symbols correspond to the units of the respective maps (e.g., places, navigation nodes, areas, objects, and landmarks) and the relations that hold between them (e.g., adjacency, inclusion, visibility). These symbols are the basis for the *conceptual map layer*. In the conceptual map, different kinds of symbolic reasoning are used to provide a human-compatible structuring and categorization of space that can be used for situated human-machine interaction.

[5] For instance, the e-puck educational robot is equipped with eight infrared (IR) proximity sensors, which measure the presence of nearby obstacles [30].

[6] The iRobot© Roomba© autonomous vacuum cleaner has the capability to find its way to a docking station by sensing the IR signals that the station emits.

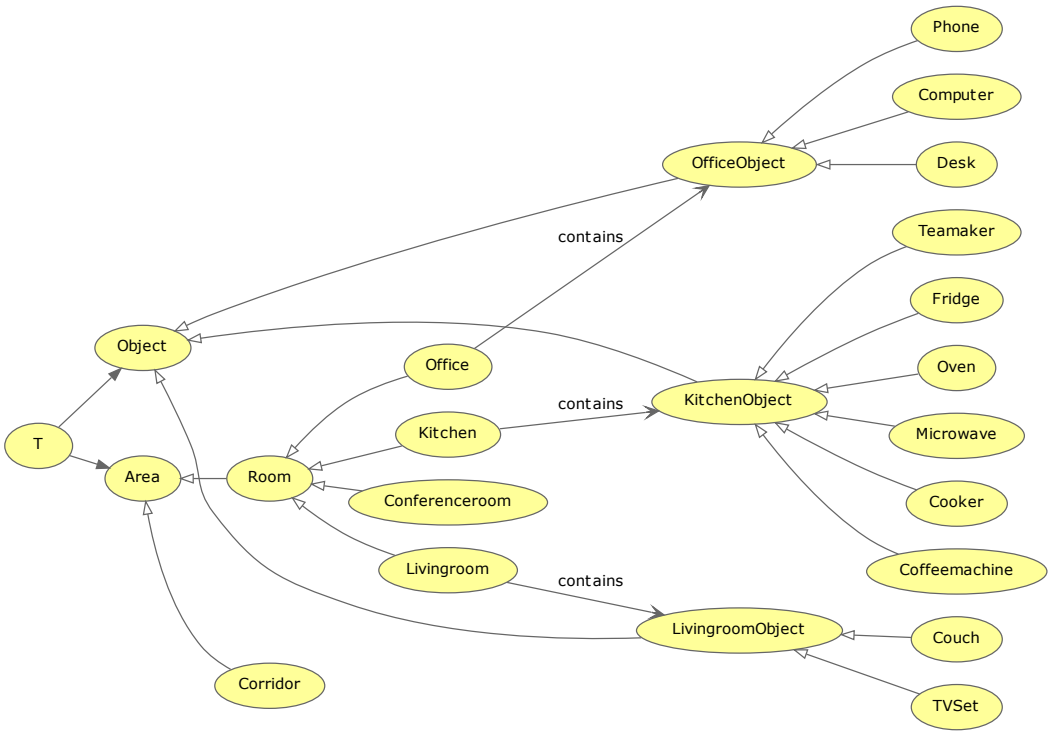



Figure 5: Illustration of a part of the commonsense ontology of an indoor office environment. Edges with hollow arrow heads denote the taxonomical subclass relation. Edges labeled with contains express that the given subclass of Room is *defined* as containing at least one instance of the pointed-to Object subclass. T stands for the universal top-level concept (i.e., $\top$ or `owl:Thing`).

With each abstraction step, the available spatial information gets coarser, while the conceptual knowledge increases. Apart from immediate adjacency of topological areas, the model is unable to derive a global structure other than containment of one portion of space in another. Specifically this means that the model cannot predict that two known areas are adjacent to each other unless their connectivity has been explicitly recognized. This corresponds, on a smaller scale, to the human performance in novel environments. Imagine the surprise when, e.g., while walking through a large furniture store, one realizes that the bathrooms are behind the bedroom closets. A similar behavior becomes apparent in [46] when the robot enters a partially explored room through a different door (thus at first believing that yet another new room has been discovered), and only afterwards arrives at a previously visited place that it knows belongs to an already known room.

In the conceptual map, information stemming from vision and dialogue is related to the spatial units generated in the lower map layers. This allows, for instance, to represent the fact that a specific object was encountered in a specific room together with the information that the human user called

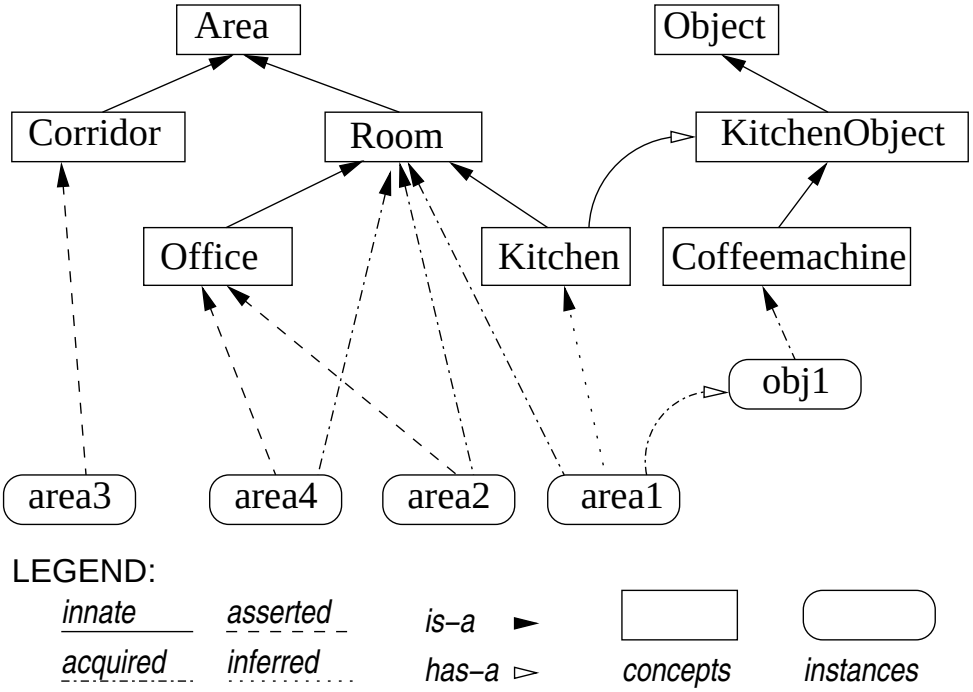



Figure 6: Combining different types of knowledge in the conceptual map.

that room "the kitchen." Internally, the conceptual map represents information about spatial areas and objects in the environment in an ontological reasoning module. It consists of a commonsense ontology of an indoor environment, which describes *taxonomies* (i.e., *subclass* relations) of room types, and couples room types to typical objects found therein through *contains* relations. Figure 5 on the previous page shows such a commonsense ontology.

These conceptual taxonomies have been handcrafted and cannot be changed online. However, instances of the concepts are added to the ontology during run-time. Using a reasoner, new knowledge can be inferred. For example, if the robot knows that it is in an area where there is a coffee machine and an oven, it can infer that it can categorize this area as a kitchen. Like this, linguistic references to areas can be generated and resolved.

# 3 Information Processing

Depending on the origin of a piece of information, we distinguish between *acquired*, *asserted*, *innate*, and *inferred* knowledge. These notions are important for the characterization of the information flow during map acquisition.

- *Acquired knowledge* is derived from the robot's own sensors, including the spatial information encoded in the lower map layers and objects recognized by a computer vision software. The information that an avatar receives from the virtual world engine is another example of acquired knowledge.

- *Asserted knowledge* is provided by another agent, for example a human tutor. It is typically given through verbal input (for example, the tutor might say "you are in the laboratory.").
- *Innate knowledge* is any kind of information that is incorporated into the system in a way that does not allow for on-line manipulation of the knowledge. In our approach, the conceptual ontology is an example of innate knowledge.
- Any piece of information that can be derived on the basis of the combination or evaluation of other information provides *inferred knowledge*, such as knowledge inferred by the Description Logic-based reasoning mechanisms in the conceptual map.

Figure 6 on the preceding page illustrates how different pieces of information are combined and processed in the conceptual map layer.

# 4 Summary

We have presented an approach to multi-layered conceptual spatial mapping for autonomous agents. It addresses the challenges of *structuring space* as well as *categorizing space*, which are prerequisites of *spatial understanding*. Since the kinds of agents we are dealing with have to operate in non-instrumented *human-oriented environments* it is crucial that they be endowed with a *human-compatible* spatial representation in order to engage in meaningful situated dialogues about spatial topics with its human user. Moreover, the presented approach allows for integration with lower-level robotic maps that provide the robot with safe and reliable navigation and control mechanisms, and which take the recent advances in robot sensing, mapping, and motion control into account.

# References


[1] Patrick Beeson, Matt MacMahon, Joseph Modayil, Aniket Murarka, Benjamin Kuipers, and Brian Stankiewicz. Integrating multiple representations of spatial knowledge for mapping, navigation, and communication. In *Interaction Challenges for Intelligent Assistants*, Papers from the AAAI Spring Symposium, Stanford, CA, USA, 2007. AAAI.

[2] Patrick Beeson, Joseph Modayil, and Benjamin Kuipers. Factoring the mapping problem: Mobile robot map-building in the Hybrid Spatial Semantic Hierarchy. *International Journal of Robotics Research*, 29(4):428–459, April 2010.

[3] Roger Brown. How shall a thing be called? *Psychological Review*, 65(1):14–21, 1958.

[4] Pär Buschka and Alessandro Saffiotti. Some notes on the use of hybrid maps for mobile robots. In *Proceedings of the 8th International Conference on Intelligent Autonomous Systems (IAS)*, Amsterdam, The Netherlands, March 2004.

[5] Eric L. Chown. Making predictions in an uncertain world: Environmental structure and cognitive maps. *Adaptive Behavior*, 7(1):17–33, December 1999.

[6] Eric L. Chown. Gateways: An approach to parsing spatial domains. In *Proceedings of the International Conference on Machine Learning Workshop on Machine Learning of Spatial Knowledge*, pages 1–6, Palo Alto, California, 2000.

[7] Eric L. Chown, Stephen Kaplan, and David Kortenkamp. Prototypes, location, and associative networks (plan): Towards a unified theory of cognitive mapping. *Cognitive Science*, 19(1):1–51, 1995.

[8] Anthony G. Cohn and Shyamanta M. Hazarika. Qualitative spatial representation and reasoning: An overview. *Fundamenta Informaticae*, 46:1–29, 2001.

[9] Kenny R. Coventry and Simon C. Garrod. *Saying, Seeing and Acting – The Psychological Semantics of Spatial Prepositions*. Essays in Cognitive Psychology. Psychology Press, 2004.

[10] Albert Diosi, Geoffrey Taylor, and Lindsay Kleeman. Interactive SLAM using laser and advanced sonar. In *Proceedings of the 2005 IEEE International Conference on Robotics and Automation (ICRA 2005)*, Barcelona, Spain, April 2005.

[11] Max J. Egenhofer and M. Andrea Rodríguez. Relation algebras over containers and surfaces: An ontological study of a room space. *Spatial Cognition and Computation*, 1(2):155–180, 1999.

[12] Deborah Estrin, David Culler, Kris Pister, and Gaurav Sukhatme. Connecting the physical world with pervasive networks. *IEEE Pervasive Computing*, 1(1):59–69, 2002.

[13] Juan-Antonio Fernández and Javier González. *Multi-Hierarchical Representation of Large-Scale Space – Applications to Mobile Robots*, volume 24 of *International Series on Microprocessor-Based and Intelligent Systems Engineering*. Kluwer Academic Publishers, Dordrecht / Boston / London, 2001.

[14] Udo Frese and Lutz Schröder. Closing a million-landmarks loop. In *Proceedings of the 2006 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2006)*, pages 5032–5039, 2006.

[15] Cipriano Galindo, Juan-Antonio Fernández-Madrigal, and Javier González. *Multiple Abstraction Hierarchies for Mobile Robot Operation in Large Environments*, volume 68 of *Studies in Computational Intelligence*. Springer Verlag, Berlin/Heidelberg, Germany, 2007.

[16] Cipriano Galindo, Alessandro Saffiotti, Silvia Coradeschi, Pär Buschka, Juan-Antonio Fernández-Madrigal, and Javier González. Multi-hierarchical semantic maps for mobile robotics. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS-05)*, pages 3492–3497, Edmonton, Canada, August 2005.

[17] Nancy L. Hazen, Jeffery J. Lockman, and Herbert L. Pick, Jr. The development of children's representations of large-scale environments. *Child Development*, 49(3):623–636, September 1978.

[18] James F. Herman and Alexander W. Siegel. The development of cognitive mapping of the large-scale environment. *Journal of Experimental Child Psychology*, 26:389–406, 1978.

[19] Stephen C. Hirtle and John Jonides. Evidence for hierarchies in cognitive maps. *Memory and Cognition*, 13:208–217, 1985.

[20] Kathryn J. Jeffery and Neil Burgess. A metric for the cognitive map: Found at last? *Trends in Cognitive Sciences*, 10(1), January 2006.

[21] Wolfgang Kainz, Max J. Egenhofer, and Ian Greasley. Modeling spatial relations and operations with partially ordered sets. *International Journal of Geographical Information Systems*, 7(3):215–229, 1993.

[22] Shanker Keshavdas. Grid based SLAM using Rao-Blackwellized particle filters. unpublished M.Sc. thesis, Heriot Watt University, Edinburgh, UK, May 2009.

[23] Bernd Krieg-Brückner, Udo Frese, Klaus Lüttich, Christian Mandel, Till Massokowski, and Robert J. Ross. Specification of an ontology for Route Graphs. In Christian Freksa, Markus Knauff, Bernd Krieg-Brückner, Bernhard Nebel, and Thomas Barkowsky, editors, *Spatial Cognition IV. Reasoning, Action, and Interaction*, volume 3343 of *Lecture Notes in Artificial Intelligence*, pages 390–412. Springer Verlag, Heidelberg, Germany, 2005.

[24] Benjamin Kuipers. *Representing Knowledge of Large-Scale Space*. PhD thesis, MIT-AI TR-418, Massachusetts Institute of Technology, Cambridge, MA, USA, May 1977.

[25] Benjamin Kuipers. The Spatial Semantic Hierarchy. *Artificial Intelligence*, 119:191–233, 2000.

[26] Benjamin Kuipers, Joseph Modayil, Patrick Beeson, Matt MacMahon, and Francesco Savelli. Local metrical and global topological maps in the Hybrid Spatial Semantic Hierarchy. In *Proceedings of the 2004 IEEE International Conference on Robotics and Automation (ICRA 2004)*, New Orleans, LA, USA, April 2004.

[27] George Lakoff and Mark Johnson. *Philosophy in the Flesh: The Embodied Mind and Its Challenge to Western Thought*. Basic Books, New York, NY, USA, 1999.

[28] Robert W. Marx. The TIGER system: Automating the geographic structure of the United States census. *Government Publications Review*, 13(2):181–201, March–April 1986.

[29] Timothy P. McNamara. Mental representations of spatial relations. *Cognitive Psychology*, 18:87–121, 1986.

[30] Francesco Mondada, Michael Bonani, Xavier Raemy, James Pugh, Christopher Cianci, Adam Klaptocz, Stéphane Magnenat, Jean-Christophe Zufferey, Dario Floreano, and Alcherio Martinoli. The e-puck, a robot designed for education in engineering. In *Proceedings of the 9th Conference on Autonomous Robot Systems and Competitions (Robotica 2009)*, pages 59–65, May 2009.

[31] Paul M. Newman, John J. Leonard, Juan D. Tardós, and José Neira. Explore and return: Experimental validation of real-time concurrent mapping and localization. In *Proceedings of the 2002 IEEE International Conference on Robotics and Automation (ICRA 2002)*, pages 1802–1809, Washington, D.C., USA, 2002.

[32] Andrzej Pronobis and Barbara Caputo. COLD: COsy Localization Database. *The International Journal of Robotics Research (IJRR)*, 28(5):588–594, May 2009.

[33] Andrzej Pronobis, Kristoffer Sjöö, Alper Aydemir, Adrian N. Bishop, and Patric Jensfelt. A framework for robust cognitive spatial mapping. In *Proceedings of the 14th International Conference on Advanced Robotics (ICAR 2009)*, Munich, Germany, June 2009.

[34] Andrzej Pronobis, Kristoffer Sjöö, Alper Aydemir, Adrian N. Bishop, and Patric Jensfelt. Representing spatial knowledge in mobile cognitive systems. Technical Report TRITA-CSC-CV 2010:1 CVAP 316, Kungliga Tekniska högskolan (KTH), CVAP/CAS, Stockholm, Sweden, March 2010.

[35] David Ribas, Pere Ridao, Juan Domingo Tardós, and José Neira. Underwater SLAM in man made structured environments. *Journal of Field Robotics*, 25(11):898–921, December 2008.

[36] M. Andrea Rodríguez and Max J. Egenhofer. Image-schemata-based spatial inferences: The container-surface algebra. In Stephen C. Hirtle and Andrew U. Frank, editors, *Spatial Information Theory: A Theoretical Basis for GIS (COSIT '97)*, volume 1329 of *Lecture Notes in Computer Science*, pages 35–52. Springer Verlag, Berlin, Germany, 1997.

[37] Eleanor Rosch. Principles of categorization. In E. Rosch and B. Lloyd, editors, *Cognition and Categorization*, pages 27–48. Lawrence Erlbaum Associates, Hillsdale, NJ, USA, 1978.

[38] Albert Stevens and Patty Coupe. Distortions in judged spatial relations. *Cognitive Psychology*, 10:422–437, 1978.

[39] Sebastian Thrun, Wolfram Burgard, and Dieter Fox. *Probabilistic Robotics*. The MIT Press, Cambridge, MA, USA, 2005.

[40] Timothy Trainor. U.S. Census Bureau geographic support: A response to changing technology and improved data. *Cartography and Geographic Information Science*, 30(2):217–223, April 2003.

[41] Shrihari Vasudevan, Stefan Gachter, Viet Nguyen, and Roland Siegwart. Cognitive maps for mobile robots – an object based approach. *Robotics and Autonomous Systems*, 55(5):359–371, May 2007.

[42] Steffen Werner, Bernd Krieg-Brückner, and Theo Herrmann. Modelling navigational knowledge by Route Graphs. In Christian Freksa, Wilfried Brauer, Christopher Habel, and Karl F. Wender, editors, *Spatial Cognition II*, volume 1849 of *Lecture Notes in Artificial Intelligence*, pages 295–316. Springer Verlag, Heidelberg, Germany, 2000.

[43] Jeremy L. Wyatt, Alper Aydemir, Michael Brenner, Marc Hanheide, Nick Hawes, Patric Jensfelt, Matej Kristan, Geert-Jan M. Kruijff, Pierre Lison, Andrzej Pronobis, Kristoffer Sjöö, Danijel Skočaj, Alen Vrečko, Hendrik Zender, and Michael Zillich. Self-understanding & self-extension: A systems and representational approach. *IEEE Transactions on Autonomous Mental Development*, under submission.

[44] Hendrik Zender. Learning spatial organization through situated dialogue. unpublished diploma thesis, Saarland University, Saarbrücken, Germany, August 2006.

[45] Hendrik Zender and Geert-Jan M. Kruijff. Multi-layered conceptual spatial mapping for autonomous mobile robots. In Holger Schultheis, Thomas Barkowsky, Benjamin Kuipers, and Bernhard Hommel, editors, *Control Mechanisms for Spatial Knowledge Processing in Cognitive / Intelligent Systems – Papers from the AAAI Spring Symposium*, Technical Report SS-07-01, pages 62–66, Menlo Park, CA, USA, March 2007. AAAI, AAAI Press.

[46] Hendrik Zender, Geert-Jan M. Kruijff, Kristoffer Sjöö, Alper Aydemir, Patric Jensfelt, Marc Hanheide, and Nick Hawes. Autonomous semantic-driven indoor exploration (report). CogX report, July 2010.

[47] Hendrik Zender, Óscar Martínez Mozos, Patric Jensfelt, Geert-Jan M. Kruijff, and Wolfram Burgard. Conceptual spatial representations for indoor mobile robots. *Robotics and Autonomous Systems*, 56(6):493–502, June 2008.

# A Realistic Benchmark for Visual Indoor Place Recognition ⋆

A. Pronobis [a,*] B. Caputo [b] P. Jensfelt [a] H.I. Christensen [c]

[a] *Centre for Autonomous Systems, The Royal Institute of Technology, SE-100 44 Stockholm, Sweden*

[b] *IDIAP Research Institute, 1920 Martigny, Switzerland EPFL, 1015 Lausanne, Switzerland*

[c] *College of Computing, Georgia Institute of Technology Atlanta, GA 30332-0760, USA*

**Abstract**

An important competence for a mobile robot system is the ability to localize and perform context interpretation. This is required to perform basic navigation and to facilitate local specific services. Recent advances in vision have made this modality a viable alternative to the traditional range sensors and visual place recognition algorithms emerged as a useful and widely applied tool for obtaining information about robot's position. Several place recognition methods have been proposed using vision alone or combined with sonar and/or laser. This research calls for standard benchmark datasets for development, evaluation and comparison of solutions. To this end, this paper presents two carefully designed and annotated image databases augmented with an experimental procedure and extensive baseline evaluation. The databases were gathered in an uncontrolled indoor office environment using two mobile robots and a standard camera. The acquisition spanned across a time range of several months and different illumination and weather conditions. Thus, the databases are very well suited for evaluating the robustness of algorithms with respect to a broad range of variations, often occurring in real-world settings. We thoroughly assessed the databases with a purely appearance-based place recognition method based on Support Vector Machines and two types of rich visual features (global and local).

*Key words:* Visual place recognition, Robot topological localization, Standard robotic benchmark

## 1 Introduction

A fundamental competence for an autonomous agent is to know its position in the world. Providing mobile robots with abilities to build an internal representation of space and obtain robust information about their location therein can be considered as one of the most urgent problems. The topic is vastly researched. This resulted, over the years, in a broad range of approaches spanning from purely metric [27,18,63], to topological [59,58,17], and hybrid [54,12]. As robots break down the fences and start to interact with people [64] and operate in large-scale environments [17,58], topological models are gaining popularity for augmenting or replacing purely metric space representations. In particular, the research on topological mapping has pushed methods for place recognition. Scalability, loop closing, and the kidnapped robot problem have been at the forefront of the issues to be addressed.

Traditionally, sonar and/or laser have been the sensory modalities of choice for place recognition and topological localization [42,38]. The assumption that the world can be represented in terms of two dimensional geometrical information allowed for many practical implementations. Yet, the inability to capture many aspects of complex realistic environments leads to the problem of perceptual aliasing [29], and greatly limits the usefulness of purely geometrical methods. Recent advances in vision have made this modality emerge as a natural and viable solution. Vision provides richer sensory input allowing for better discrimination. It opens new possibilities for building cognitive systems, actively relying on semantic context. Not unimportant is the cost effectiveness, portability and popularity of visual sensors. As a result, this research line is attracting more and more attention, and several methods have been proposed using vision alone [56,48,51,17], or combined with more traditional range sensors [28,53,50].

In spite of large progress, vision-based localization still represents a major challenge. First of all, visual information tends to be noisy and difficult to interpret. The visual appearance of places varies in time because of illumination changes (day and night, artificial light on and off) and because of human activities (furniture moved around, objects being taken out of drawers, and so on). Thus, the solutions must be highly robust, provide good generalization abilities and in general be adaptive. Additionally, the application puts strong

⋆ A preliminary version of the experimental evaluation reported in this work was presented in [49]: A. Pronobis, B. Caputo, P. Jensfelt, and H. I. Christensen. A discriminative approach to robust visual place recognition. In *Proceedings of IROS'06*.

* Corresponding author.
*Email addresses:* pronobis@csc.kth.se (A. Pronobis), bcaputo@idiap.ch (B. Caputo), patric@csc.kth.se (P. Jensfelt), hic@cc.gatech.edu (H.I. Christensen).

constraints on the computational complexity, and the increased resolution and dimensionality of the visual data still constitutes a problem.

The fact that so many different parameters influence the accuracy of a vision-based localization system is another challenge itself, especially burdensome at the design stage. As the results depend greatly on the choice of training and test input data, which are unstable over time, it is hard to measure the influence of the different parameters on the overall performance of the system. For the same reason, it becomes nearly impossible to compare fairly solutions which are usually evaluated in different environments, under different conditions, and with different assumptions. This is a major obstacle slowing down progress in the field. There is a need for standardized benchmarks and databases which would allow for fair comparisons, simplify the experimental process and boost development of new solutions.

Databases are heavily exploited in the computer vision community, especially for object recognition and categorization [25,4,3]. As the community acknowledges the need for benchmarking, a lot of attention is directed towards designing new datasets, reflecting the increasing capabilities of visual algorithms [45]. Also in robotics, research on Simultaneous Localization and Mapping (SLAM) makes use of several publicly available datasets [26,40]. Still, no database emerged as a standard benchmark for visual place recognition applied to robot localization.

This paper aims at filling this gap, and presents a benchmark consisting of two different image databases gathered in the same indoor environment. The databases are augmented with an experimental procedure as well as extensive baseline evaluation. The datasets were carefully designed and later annotated. Three different imaging devices were used for acquisition (two mobile robot platforms and a standard camera), resulting in data of different characteristics and quality. In order to create a realistic and challenging test bed, the acquisition process was performed in an uncontrolled typical office environment, under various illumination and weather conditions (sunny, cloudy, night), and over a significant span of time. All of this makes the databases very well suited for evaluating robustness of visual place recognition algorithms, applied to the problem of robot topological localization, in presence of different types of variations often occurring in real-world indoor settings.

An important component when providing the community with a new collection of data is to provide a baseline evaluation that illustrates the nature of the dataset (see Section 5.1 for explanation). We thoroughly assessed the databases with a purely appearance-based place recognition method. The method uses two types of image descriptors, local and global, in order to extract rich visual information. Both descriptors have shown remarkable performances, coupled with computational efficiency on challenging object recog-

nition scenarios [31,30]. The classification step is performed using Support Vector Machines [16] and specialized kernels are used for each descriptor. Results show that the method is able to recognize places with high precision and robustness under varying illumination conditions, even when training on images from one camera device and testing on another.

The rest of the paper is organized as follows: after a review of related literature (Section 2), we discuss the problem and challenges we addressed with the benchmark (Section 3). Then, Section 4 gives a detailed description of the data acquisition process and scenario and presents the acquisition results. Finally, the algorithm used for the baseline evaluation as well as the experimental procedure are described in Section 5, and the experimental results are given in Section 6. The paper concludes with a summary (Section 7).

## 2 Related work

Place recognition and topological localization are vastly researched topics in the robotic community, where vision and laser range sensors are usually the privileged modalities. Although laser-based solutions have proven to be successful for certain tasks [38], their limitations inspired many researchers to turn towards vision which nowadays becomes tractable in real-time applications. The available methods employ either perspective [56,52,20] or omnidirectional cameras [23,9,59,35,6,39,60]. The main differences between the approaches relate to the way the scene is perceived, and thus the method used to extract characteristic features from the scene. Landmark-based techniques make use of either artificial or natural landmarks in order to extract information about a place. Mata et al. [34] proposed a system able to interpret information signs through its ability to read text and recognize icons. Visually distinctive image regions were also used as landmarks [51]. Other solutions employed mainly local image features such as SIFT [31,6,48], SURF [8,39,60], also using the bag-of-words approach [20,22,17], or representation based on information extracted from local patches using Kernel PCA [52]. Global features are also commonly used for place recognition. Torralba et al. [57,56,55] suggested to use a representation called the "gist" of the scene, which is a vector of principal components of outputs of a bank of spatially organized filters applied to the image. Other approaches use color histograms [59,9], gradient orientation histograms [11], eigenspace representation of images [23], or Fourier coefficients of low frequency image components [35]. Recently, several authors observed that robustness and efficiency of the recognition system can be improved by combining information provided by both types of visual cues (global and local) [48,51,62]. Although vision-based localization methods are now commonly applied, it remains extremely difficult to compare the different approaches, as the evaluations presented by the authors usually follow different procedures

and are performed on different sets of visual data.

There are a number of heavily used standard databases in robotics and computer vision. In robotics, these databases are used mainly for testing algorithms for simultaneous localization and mapping (SLAM) [26,40] and mostly contain odometry and range sensor data. In case of the computer vision community, the effort concentrated on creating standard benchmarks for such problems as object [25,4,45], action [33], scene [3], or texture recognition and categorization [2]. The MIT-CSAIL Database of Objects and Scenes [3] is a notable exception as it provides several image sequences acquired in both indoor and outdoor environments and was used to evaluate performance of a visual place recognition system.

This paper makes an important contribution by providing annotated data from visual and laser range sensors together with an experimental procedure that can be followed in order to evaluate place recognition and localization systems. In contrast to the previously available benchmarking solutions, the databases contain several sets of images and image sequences acquired in the same environment under various conditions and over a significant span of time. This makes them perfect for evaluating robustness of the algorithms under dynamic variations that often occur in realistic settings. The introduction of standard benchmark databases has made an impact on the research on such problems as object categorization or simultaneous localization and mapping (SLAM), allowing different methods to be more fairly compared in the same scenario. The authors hope that the benchmark proposed in this paper will similarly influence the research on visual place recognition in the context of mobile robot localization.

## 3 Design strategy

This section defines and characterizes the problem that we address with the benchmark (Section 3.1) and analyzes the difficulties and open challenges in visual place recognition that have to be considered in a realistic scenario (Section 3.2).

### *3.1 Problem Statement*

Let us begin with a brief definition of a place and the place recognition problem that we will use throughout this paper. A place can be regarded as a usually nameable segment of a real-world environment distinguished due to different functionality, appearance or artificial boundary. In view of this definition, the

place recognition or identification problem can be characterized as follows. Given a set of training sensory data, captured in each of the considered places, build models of the places reflecting their inherent properties. Next, when presented with new test data, unavailable during training, acquired in one of the same places, identify the place where the acquisition was performed (e.g. Barbara's office) based on the knowledge encoded in the models. This is different from the problem of place categorization where the task is to classify test data captured in a novel place as belonging to one of the place categories (e.g. an office). As the partition of space into different places can be based on several criteria, here we consider a supervised scenario where the algorithm has to distinguish between five areas of different functionality, selected by a teacher.

This benchmark is designed to test the performance of a visual place recognition system on images acquired within an indoor office environment. As the primary scenario, we consider the case where a place recognition system is used to provide a mobile robot with information about its location. For this reason, part of the data presented in this paper was acquired using cameras mounted on mobile robot platforms. While designing the benchmark, we concentrated on testing the ability of a visual recognition system to identify a place based on one image only. This makes the problem harder, but also makes it possible to perform global localization where no prior knowledge about the position is available (e.g. in case of the kidnapped robot problem). Spatial or temporal filtering can be used together with the presented methods to enhance performance.

We concentrate on indoor environments, since in the considered scenario, they play a crucial role, being typical spaces for the interaction between humans and service robots or robotic assistants [64]. At the same time, office environments, just like home environments, constitute an important class of indoor spaces for robotic companions. In this benchmark, our aim is to provide datasets and experimental procedures that will allow for evaluating robustness of place recognition systems based on different types of visual cues to typical variations that occur in an indoor environment for the considered scenario. These include illumination changes, variations introduced by human activity and viewpoint changes. As a consequence, instead of providing datasets spanning over a very large portion of space, we provide image sequences acquired over a time span of several months, under various illumination conditions and using different devices. The proposed evaluation framework should allow for concluding that an algorithm robust to the variations captured in the benchmark data will be robust to similar types of variations within other indoor office environments.

The benchmark is designed for evaluating vision-based methods. We choose vision as sensory modality for several reasons. First, the visual sensor is very rich and, although also very noisy, provides great descriptive capabilities. This

is crucial in indoor environments where other sensors, such as a laser range finder, suffer from the problem of perceptual aliasing (different places look the same [29]). Furthermore, the visual appearance of places encodes information about their semantics, which plays a major role in enabling systems to interact with the environment. Finally, in the era of cheap portable devices equipped with digital cameras, it is also one of the most affordable and commonly available solutions.

### *3.2 Challenges*

Recognizing indoor places based on their visual appearance is a particularly challenging task. First of all, in case of indoor environments, there is no obvious spatial layout that once observed could be used to distinguish between different places. Moreover, viewpoint variations cause the visual sensor to capture different aspects of the same place, which often can only be learned if enough training data are provided. At the same time, real-world environments are usually dynamic and their appearance changes over time. The visual recognition system must be robust to variations introduced by changing illumination as well as human activity. For a visual sensor, the same room might look different during the day, during sunny weather, under direct natural illumination, and at night with only artificial light turned on. Moreover, if the environment is being used, the fact that people appear in the images, objects are being moved or furniture relocated may greatly influence the performance of the system. All these issues were taken into consideration while designing this benchmark in order to create a realistic test bed.

## 4 Data Acquisition

Based on the analysis of the problem presented in the previous section, we carefully designed and acquired two databases comprising images captured in the same indoor environment, but using different devices: the INDECS (INDoor Environment under Changing conditionS) database [47] and the IDOL (Image Database for rObot Localization) database [32]. This section describes the resulting data acquisition procedure. In case of INDECS, we acquired images of the environment from a fixed set of points using a standard camera mounted on a tripod. The resolution of the images is high; this makes this database suitable for context-based object recognition. The IDOL database, instead, consists of image sequences recorded using two mobile robot platforms equipped with perspective cameras, and thus is well suited for experiments with robot localization. All three devices are shown in Fig. 1. The databases represent a different approach to the problem and can be used to

(a) Minnie (b) Camera (c) Dumbo

Fig. 1. Devices employed in the acquisition: the two mobile robot platforms "Minnie" (a) and "Dumbo" (c) as well as the standard camera on a tripod (b).

analyze different properties of a place recognition system. The acquisition was performed under several different illumination settings and over a significant span of time. Both databases are publicly available and can be downloaded from `http://www.csc.kth.se/~pronobis`.

The rest of the section is organized as follows: Section 4.1 presents the acquisition scenario, as to say the environment where both databases were acquired. Then, Section 4.2 provides a description of the INDECS database, and Section 4.3 gives detailed information on the robot platforms and IDOL. Finally, we perform an analysis of the obtained data in Section 4.4.

### *4.1 Acquisition Scenario*

The acquisition was conducted within a five room subsection of a larger office environment of the Computer Vision and Active Perception Laboratory at the Royal Institute of Technology in Stockholm, Sweden. Each of the five rooms represents a different type of functional area: a one-person office, a two-persons office, a kitchen, a corridor, and a printer area (in fact a continuation of the corridor). The function that a room fulfills determines the furniture, objects, and activity that is likely to be found there. Places like the corridor, the printer area and the kitchen can be regarded as public which implies that various people may be present. On the other hand, offices were imaged usually empty or with their owners at work. In the corridor and the printer area, furniture

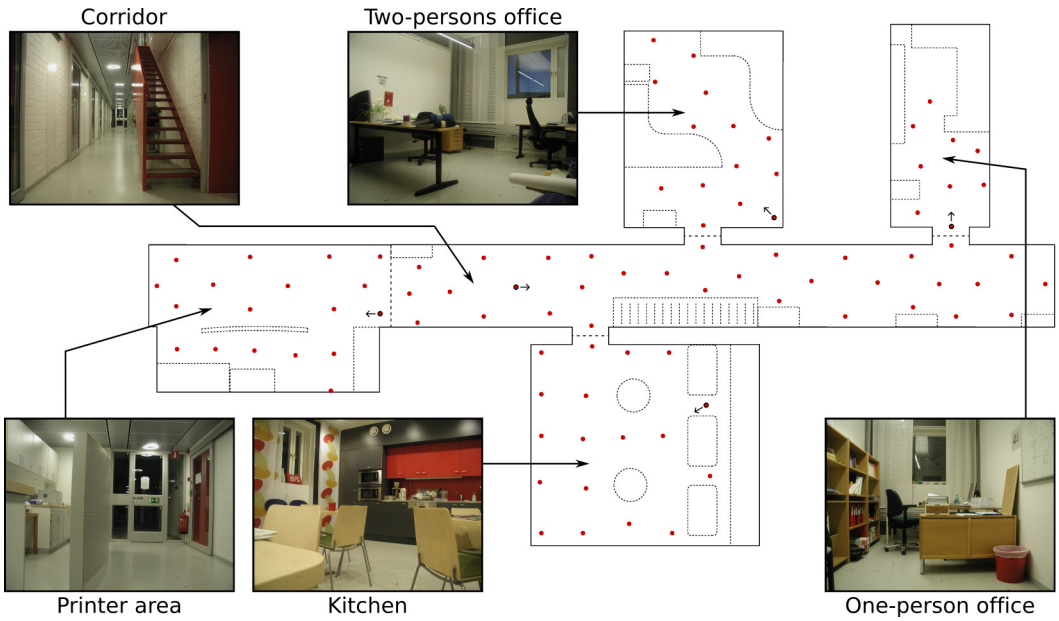


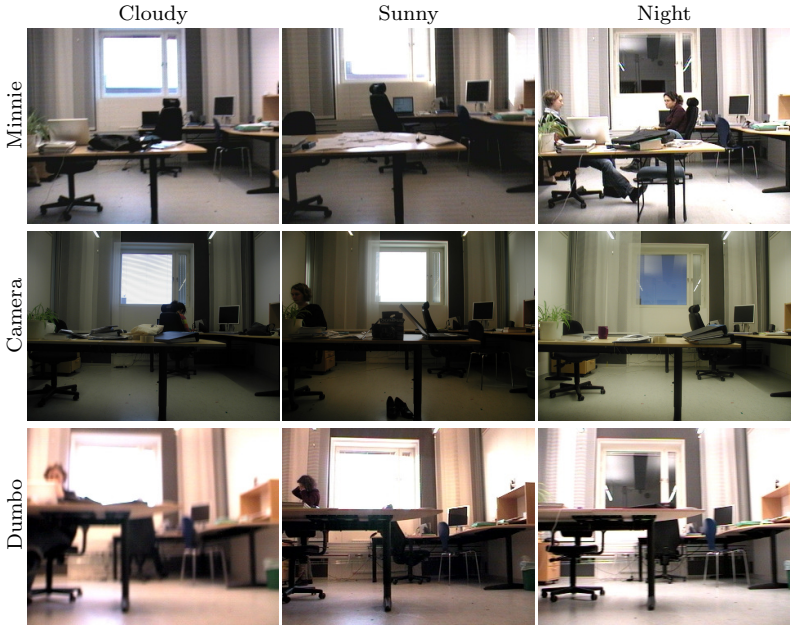

Fig. 2. A general map of the part of the office environment that was imaged during acquisition of the INDECS and IDOL databases. Boundaries between the five rooms were marked with dashed lines. Dotted lines were used to draw an approximate outline of furniture. Moreover, the location of points at which the tripod was placed while recording the INDECS database were marked. The pictures are taken from the database and show the interiors of the five rooms. The small arrows were used to indicate the viewpoints at which the presented pictures were taken.

is mostly fixed and objects are less moveable. As a result, these areas were less susceptible to variations caused by human activity in comparison to the kitchen or the offices, where furniture (e.g. chairs) is relocated more often and objects (e.g. cups, laptops etc.) are frequently moved.

The rooms are physically separated by sliding glass doors. The printer area is an exception and was treated as a separate place only due to its different functionality (the border between the corridor and the printer area was arbitrarily defined). The laboratory contains additional rooms which were not taken into consideration while creating the database. However, because of the glass door, other parts of the environment can still be visible in the images. Examples of pictures showing the interior of each room as well as a general map of the environment are presented in Fig. 2.

As already mentioned, the visual data were acquired with three different devices. In each case, the appearance of the rooms was captured under three different illumination and weather conditions: in cloudy weather (natural and artificial light), in sunny weather (direct natural light dominates), and at night (only artificial light). Since all the rooms have windows, the influence of natural illumination was significant. The image acquisition was spread over a period of time of three months, for the INDECS database, and over two weeks for the


(a) Corridor (b) Kitchen




(c) Two-persons office

Fig. 3. Example pictures taken from the INDECS and IDOL databases acquired with the camera and the two robot platforms under various illumination conditions. The pictures show the influence of illumination (especially (a) and (c)) and illustrate the differences between images acquired in a cluttered environment using different devices (b). Additional variability caused by natural activities in the rooms is also apparent (presence of people, relocated objects and furniture).

IDOL database. Additionally, the INDECS database was acquired ten months before the experiments with the robots. In this way, we captured the visual variability that occurs in realistic environments due to varying illumination and natural activities in the rooms. Fig. 3 presents a comparison of images taken under different illumination conditions and using various devices.

−90° −60° −30° 0° 30° 60°

Fig. 4. Pictures from the INDECS database taken from several angles at the same location in the two-persons office.

### *4.2 The INDECS database*

The INDECS database consists of pictures of the environment described above, gathered from different viewpoints using a standard camera mounted on a tripod. We marked several points in each room (approximately one meter apart) where we positioned the camera for each acquisition. The rough positions of all points are shown on the map in Fig. 2. The number of points changed with the dimension of the room, from a minimum of 9 for the one-person office to a maximum of 32 for the corridor. At each location we acquired 12 images, one every 30°, even when the tripod was located very close to a wall or furniture. Examples of images taken at the same location and from several angles are presented in Fig. 4. Images were acquired using an Olympus C-3030ZOOM digital camera and the height of the tripod was constant and equal to 76 cm. All images in the INDECS database were acquired with a resolution of 1024x768 pixels, the auto-exposure mode enabled, flash disabled, the zoom set to wide-angle mode, and the auto-focus enabled. In this paper, the INDECS images were subsampled to 512x386 before being used in the experiments. The images were labeled according to the position of the point where the acquisition happened. As a consequence, images taken, for example, from the corridor but looking into a room are labeled as the corridor. The images were acquired across a time span of three months and under varying illumination conditions (sunny, cloudy and night). For each illumination setting, we captured one full set of images. In total, there are 3264 images (324 for the one-person office, 492 for the two-persons office, 648 each for the kitchen and the printer area, and 1152 for the corridor) in the INDECS database.

### *4.3 The IDOL database*

The IDOL database was acquired using cameras on two mobile robot platforms. Both robots, the PeopleBot Minnie and the PowerBot Dumbo, were equipped with a pan-tilt-zoom Canon VC-C4 camera, a SICK laser range finder, and wheel encoders. However, as it can be seen from Fig. 1, the cameras were mounted at different height. On Minnie, the camera was 98cm above the floor, whereas on Dumbo it was 36cm. Furthermore, the camera on Dumbo was tilted up approximately 13°, to reduce the amount of floor captured in the

images. The selected positions of the cameras result in different characteristics of the environment being captured in the images. Due to the low placement of the camera on Dumbo, the captured images are less susceptible to variations introduced by human activity in the environment and direct sunlight coming through the windows. At the same time, the camera on Minnie was able to capture the appearance of objects located on the desks and provide more information about the semantics of a place. All images were acquired with a resolution of 320x240 pixels, with the zoom fixed to wide-angle (roughly 45° horizontal and 35° vertical field of view), the auto-exposure and the auto-focus modes enabled.

We followed the same procedure during image acquisition with both robot platforms. Each robot was manually driven (average speed around 0.3-0.35m/s) through each of the five rooms while continuously acquiring images at the rate of five frames per second. The path was roughly planned so that the robots could capture the visual appearance of all the rooms. For the different illumination conditions (sunny, cloudy, night), the acquisition procedure was performed twice, resulting in two image sequences acquired one after another giving a total of six sequences for each robot platform across a span of over two weeks. Each of the image sequences in the database is accompanied by laser scans and odometry data. Due to the manual control, the path of the robot was slightly different for every sequence. Examples of paths are presented in Fig. 7, 8, and 9. Each image sequence consists of 1000-1300 frames. To automate the process of labeling the images for the supervision, the robot pose was estimated during the acquisition process using a laser based localization method [21]. Again, each image was labeled as belonging to one of the five rooms based on the position from where it was taken.

### *4.4 Acquisition Results*

Examples illustrating the properties of images that can be found in both databases are given in Fig. 3. First of all, we can observe a significant influence of illumination. The appearance of the rooms is affected by highlights, shadows and reflections, especially in case of strong direct sunlight. Moreover, the fact that the auto-exposure mode was on, resulted in a lower contrast in the informative parts of images, when the camera was directed towards a bright window in sunny weather. At the same time, the conditions observed during cloudy weather were much more stable and could be seen as intermediate between those during sunny weather and at night. A second important type of variability was introduced by human presence and activities. In some cases, people partially occluded the view. Furthermore, the fact that the environment was observed for some time, allowed to capture different configurations of furniture or objects placed on the desks or kitchen tables. The fact that ob-

jects could be observed in the images makes it possible to use the database in more complex scenarios where place recognition and object recognition complement each other e.g. by contextual priming [56,55] (especially in case of the high resolution images in the INDECS database). Finally, we can compare the images acquired using the three different devices. We see that each device captures different aspects of the same environment, mainly due to the variations in viewpoint caused by the different heights of the cameras. The influence of viewpoint is substantial, especially for cluttered scenes, when the camera was close to the furniture.

For both databases, the environment was observed from multiple viewpoints. For INDECS, the viewpoints are stable over different weather conditions, but the appearance of the rooms is almost fully captured as the images were taken in 12 directions. In case of IDOL, we observe changes in viewpoint due to manual control of the robot, but since the robot was driven in a particular direction, parts of the environment might not be observed. As previously mentioned, labelling was based on the position of the camera rather than contents of the images, and acquisition was performed even close to walls or furniture. As a result, both databases contain difficult cases, where the contents of the image is either non-informative or is weakly associated with the label.

To summarize, despite the fact that the acquisition was performed in a relatively small environment (consisting of 5 different rooms), there are several types of variability captured which pose a challenge to a recognition system. These range from different acquisition conditions to large viewpoint variations across the devices. Moreover, the acquisition procedure was carefully designed, and each single dataset offers different, but usually well specified, type of variability. As a result, the influence of different factors on the accuracy of the system can be isolated and precisely measured. The relatively small environment does not allow for concluding that a system evaluated on the data will offer similar absolute performance in a different environment. However, since the data capture the influence of a large amount of variations on the appearance of a standard office environment, we can expect that an algorithm robust to those variations will be robust to similar types of variations within other indoor office environments.

## 5 Baseline Evaluation

This section presents the visual place recognition system with which we assessed the INDECS and IDOL databases. We applied a fully supervised, appearance-based method. It assumes that each room is represented, during training, by a collection of images capturing its visual appearance under different viewpoints, at a given time and illumination. During testing, the al-

gorithm is shown images of the same rooms, acquired under roughly similar viewpoints but possibly under different illumination conditions and after some time (where the time range goes from some minutes to several months). The goal is to recognize correctly each single image seen by the system. The method is based on a large-margin discriminative classifier, namely the Support Vector Machines (SVMs) [16] and two different image representations. We use global and local image features, and we combine them with SVMs through specialized kernels. As a result, the recognition process always consists of two steps: feature extraction and classification.

In the rest of this section, we first motivate the decision to provide a baseline evaluation with the presented datasets (Section 5.1). Then, we describe the employed image representations (Section 5.2) and the classifier (Section 5.3). Finally, we explain the procedure followed in our experiments (Section 5.4).

### *5.1 Motivation*

An important component when providing the community with a new collection of data is to give a quantitative measure of how hard the database is. Benchmark databases have become a very popular tool in several research communities during the last years [25,33], because they provide at the same time an instrument to develop new state of the art algorithms, and a way to call attention on a research topic. When a database is used for developing a new algorithm, it is extremely useful to be able to compare the obtained results with those obtained by some other established technique: this permits to understand what are the advantages of the new method over existing approaches. At the same time, presenting a new corpus together with a baseline evaluation helps the community to quickly identify the open challenges of the problem and therefore concentrate there their research efforts. While often the baseline evaluation consists of a newly developed method, very often it is a well known, off the shelf solution: again, the goal of a baseline evaluation is not that of presenting new theory, but to provide a quantitative evaluation of how challenging the new dataset is, coupled with a well defined experimental protocol.

The computer vision community has been traditionally very open to the introduction of publicly available databases [25,33] and associated benchmark challenges [4]. These two tools, combined together, have heavily contributed to set the research agenda of the last years. The robotics community has recently started to acknowledge the value and power of such collections, as it is witnessed by several successful benchmark evaluations [5,1].

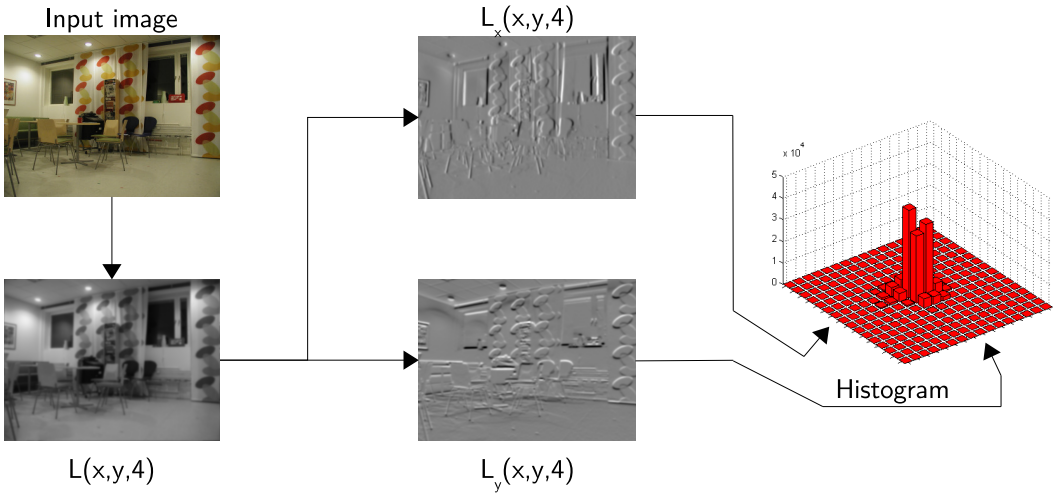



Fig. 5. The process of generating multi-dimensional receptive field histograms shown on the example of the first-order derivatives computed at the same scale $t = 4$ from the illumination channel.

### *5.2 Feature Extraction*

The feature extraction step aims at providing a representation of the input data that minimizes the within-class variability while at the same time maximizing the between-class variability. Additionally, this representation is usually more compact than raw input data and therefore allows to reduce the computational load imposed by the classification process. Features can be derived from the whole image (global features) or can be computed locally, based on its salient parts (local features).

As environments can be described differently, depending on the considered scale, scale-space theory appears as a suitable framework for deriving effective representations here. Following this intuition, we chose to use two scale-space theory based features, one global (Composed Receptive Field Histograms, CRFH [30]) and one local (Scale Invariant Feature Transform, SIFT [31]). The rest of the section describes briefly the two approaches.

#### *5.2.1 Global Features: Compose Receptive Field Histograms*

CRFH is a multi-dimensional statistical representation of the occurrence of responses of several image descriptors applied to the image. This idea is illustrated in Fig. 5. Each dimension corresponds to one descriptor and the cells of the histogram count the pixels sharing similar responses of all descriptors. This approach allows to capture various properties of the image as well as relations that occur between them.

Multi-dimensional histograms can be extremely memory consuming and computationally expensive if the number of dimensions grows. For example, a 9-dimensional histogram with 16 quantization levels per dimension contains approximately $7 \cdot 10^{10}$ cells. In [30], Linde and Lindeberg suggest to exploit the fact that most of the cells are usually empty, and to store only those that are non-zero. The histogram can be stored in a sparse form as an array $[(c_1, v_1), (c_2, v_2), \ldots, (c_n, v_n)]$, where $c_i$ denotes the index of the cell containing the non-zero value $v_i$. This representation allows not only to reduce the amount of memory required, but also to perform operations such as histogram accumulation and comparison efficiently. For our experiments, we built multi-dimensional histograms using combinations of several image descriptors, applied to the scale-space representation at various scales, namely: first- and second-order Gaussian derivatives, gradient magnitude, Laplacian and Hessian determinant applied to both intensity and color channels.

#### *5.2.2 Local Features: Scale Invariant Feature Transform*

The idea behind *local features* is to represent the appearance of an image only around a set of characteristic points known as the *interest points*. The similarity between two images is then measured by solving the correspondence problem. Local features are known to be robust to occlusions, as the absence of some interest points does not affect the features extracted from other local patches.

The process of local feature extraction consists of two stages: *interest point detection* and *description*. The interest point detector identifies a set of characteristic points in the image that could be re-detected even in spite of various transformations (e.g. rotation and scaling) and variations in illumination conditions. The role of the descriptor is to extract robust features from the local patches located at the detected points.

In this paper, we used the scale, rotation, and translation invariant Harris-Laplace detector [36] and the commonly used SIFT descriptor [31]. Comparisons of local descriptors and interest point detectors, presented in [37], show that both algorithms are highly reliable. Moreover, the SIFT descriptor has shown to perform well for object classification ([19]) and mobile robot localization ([6,20]).

### *5.3 Classification: Support Vector Machines*

The choice of the classifier is the second key ingredient for an effective visual place recognition system. In this paper, we chose Support Vector Machines (SVMs) based on their state-of-the-art performances in several visual recog-

nition domains [41,13,7]. The rest of this section reviews briefly the theory behind the algorithm, and describes our choices for the kernel function. We refer the readers to [16] for a thorough introduction to the subject.

### *5.3.1 Linear SVM*

Consider the problem of separating a set of training data $(\boldsymbol{x}_1, y_1), ...(\boldsymbol{x}_m, y_m)$ into two classes, where $\boldsymbol{x}_i \in \Re^N$ is a feature vector and $y_i \in \{-1, +1\}$ its class label. Assuming that the two classes can be separated by a hyperplane $\boldsymbol{w} \cdot \boldsymbol{x} + b = 0$, then the optimal hyperplane will be the one with maximum distance to the closest points in the training set. The optimal values for $\boldsymbol{w}$ and $b$ can be found by solving a constrained minimization problem via Lagrange multipliers, resulting in a classification function

$$f(\boldsymbol{x}) = \text{sgn}\left(\sum_{i=1}^{m} \alpha_i y_i \boldsymbol{x}_i \cdot \boldsymbol{x} + b\right), \tag{1}$$

where $\alpha_i$ and $b$ can be found efficiently using the Sequential Minimal Optimization (SMO, [43]) algorithm. The $\boldsymbol{x}_i$ with $\alpha_i \neq 0$ are called *support vectors*.

### *5.3.2 Non-linear SVM and Kernel Functions*

To obtain a nonlinear classifier, one maps the data from the input space $\Re^N$ to a higher dimensional feature space $\mathcal{H}$ by $\boldsymbol{x} \rightarrow \Phi(\boldsymbol{x}) \in \mathcal{H}$, such that the mapped data points of the two classes are linearly separable in the feature space. Assuming there exists a kernel function $K$ such that $K(\boldsymbol{x}, \boldsymbol{y}) = \Phi(\boldsymbol{x}) \cdot \Phi(\boldsymbol{y})$, a nonlinear SVM can be constructed by replacing the inner product $\boldsymbol{x}_i \cdot \boldsymbol{x}$ by the kernel function $K(\boldsymbol{x}_i, \boldsymbol{x})$ in Eqn. (1). This corresponds to constructing an optimal separating hyperplane in the feature space.

The choice of the kernel function is a key ingredient for the good performance of SVMs; based on results reported in the literature, we chose in this paper the $\chi^2$ *kernel* [15] for global features and the *match kernel* [61] for local features.

The $\chi^2$ kernel belongs to the family of exponential kernels, and is given by

$$K(\boldsymbol{x}, \boldsymbol{y}) = \exp\left\{-\gamma \chi^2(\boldsymbol{x}, \boldsymbol{y})\right\}, \qquad \chi^2(\boldsymbol{x}, \boldsymbol{y}) = \sum_i \frac{||x_i - y_i||^2}{||x_i + y_i||}. \tag{2}$$

The match kernel is given by [61]

$$K(\boldsymbol{L}_h, \boldsymbol{L}_k) = \frac{1}{n_h} \sum_{j_h=1}^{n_h} \max_{j_k=1,\ldots,n_k} \left\{ K_l(\boldsymbol{L}_h^{j_h}, \boldsymbol{L}_k^{j_k}) \right\}, \tag{3}$$

where $\boldsymbol{L}_h, \boldsymbol{L}_k$ are local feature sets and $\boldsymbol{L}_h^{j_h}, \boldsymbol{L}_k^{j_k}$ are two single local features. The sum is always calculated over the smaller set of local features and only some fixed amount of best matches is considered in order to exclude outliers. The local feature similarity kernel $K_l$ can be any Mercer kernel. We used the RBF kernel based on the Euclidean distance for the SIFT features:

$$K_l(\boldsymbol{L}_h^{j_h}, \boldsymbol{L}_k^{j_k}) = \exp\left\{ -\gamma ||\boldsymbol{L}_h^{j_h} - \boldsymbol{L}_k^{j_k}||^2 \right\}. \tag{4}$$

The match kernel was introduced in [61], and despite the claim in the paper, it is not a Mercer kernel [10]. Still, it can be shown that it statistically approximates a Mercer kernel in a way that makes it a suitable kernel for visual applications [10]. On the basis of this finding, and of its reported effectiveness for object categorization [41], we will use it here.

#### *5.3.3 Multi-class SVM*

The extension of SVM to multi class problems can be done mainly in two ways:

- *One-vs-all strategy.* If $M$ is the number of classes, $M$ SVMs are trained, each separating a single class from all remaining classes. The decision is then based on the distance of the classified sample to each hyperplane and the final output is the class corresponding to the hyperplane for which the distance is largest.
- *One-vs-one strategy.* In this case, $M(M-1)/2$ two-class machines are trained for each pair of classes. The final decision can then be taken in different ways, based on the $M(M-1)/2$ outputs. A popular choice is to consider as output of each classifier the class label and count votes for each class; the test image is then assigned to the class that received more votes. Another alternative is to use signed distance from the hyperplane and sum distances for each class. Other solutions based on the idea to arrange the pairwise classifiers in trees, where each tree node represents an SVM, have also been proposed [44,16].

In this paper, for efficiency reasons, we will use the pairwise approach and the voting-based method, which we found to constantly outperform the second variant in preliminary experiments (the complexity of the SVM training algorithm is approximately $O(n^2)$ and smaller training subsets of the binary classifiers make the training procedure faster).

### *5.4 Experimental Setup*

We conducted four series of experiments in order to assess thoroughly the INDECS and IDOL databases. For each series of experiments, we evaluated the performance of both local and global image representations. We divided the databases into several subsets with respect to the illumination conditions that prevailed during acquisition and the device employed. For the INDECS database, we considered three image sets, one for each illumination setting (cloudy, night, sunny). Since the IDOL database consists of 12 image sequences, we used each full sequence as a separate set. The system was always trained in a supervised fashion on one, two or three data sets and tested on a fourth different set. In order to test the limits of the underlying visual recognition algorithm, we considered each image in the test set separately, and as a final measure of performance, we used the percentage of properly recognized images. As the number of acquired images varied across rooms, the performance obtained for each place was considered separately during the experiments. The final classification rate was then computed as the average between all the rooms results. This procedure ensures that performance on each place contributes equally to the overall result, thus avoiding the biases towards areas with many acquired images, such as the corridor.

We started with a set of reference experiments, assessing the data acquired under stable illumination. To achieve this, for training and testing we used data sets acquired with the same device and under similar conditions. Next, we increased the difficulty of the problem and tested the robustness of the system to changing illumination conditions as well as to other variations that may occur in real-world environments. Training and recognition were in this case performed on data sets consisting of images captured under different illumination settings and usually on different days. The third set of experiments aimed to reveal whether a model trained on an image set acquired with one device can be useful for solving localization problem with a different device (and usually after some time). Finally, we checked whether the robustness of the recognition algorithm can be increased by providing additional training data capturing a wider spectrum of visual variability. For that, we trained the system on two or three image sets gathered under different illumination conditions. Additionally, before carrying out the benchmarks described above, we conducted a set of preliminary experiments in order to select proper kernel functions and feature extractor parameters. All the results obtained on these experiments are reported in Section 6.

For all experiments, we used our extended implementation of Support Vector Machines based on the *libsvm* software [14]. We set the value of the error penalty $C$ to be equal to 100 and we determined the kernel parameters via cross-validation.

## 6 Experimental Results

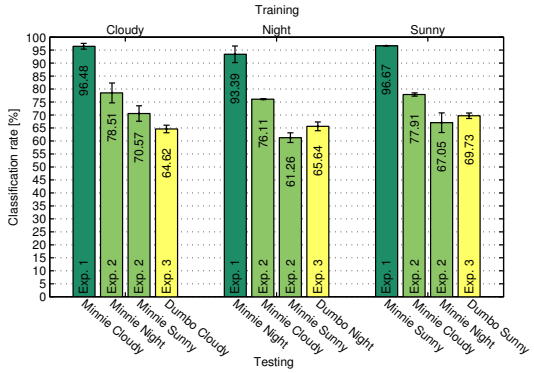
This section reports the results of the baseline evaluation of the INDECS and IDOL databases, according to the procedure described in Section 5.4. We present the results in consecutive subsections, and we give a brief summary in Section 6.5.

As described in Section 5.4, before performing the actual benchmark, we ran a set of preliminary experiments on the INDECS database, mainly using the global features (CRFH). We evaluated the performance of the multi-dimensional histograms built from a wide variety of combinations of global image descriptors listed in Section 5.2 for several scale levels and numbers of histogram bins per dimension. A comprehensive report on the obtained results can be found in [46]. The experiments revealed that the most valuable global features can be extracted using non-isotropic, derivative-based descriptors, and that chromatic cues are more susceptible to illumination variations. As a result, here we used composed receptive field histograms of six dimensions with 28 bins per dimension, computed from second order normalized Gaussian derivative filters, applied to the illumination channel at two scales. The scale levels were different for the experiments with IDOL ($\sigma = 1$ and 4) and with INDECS ($\sigma = 2$ and 8). This was motivated by the fact that the cameras mounted on the robots obtained images of lower quality, and their movement introduced additional distortions.

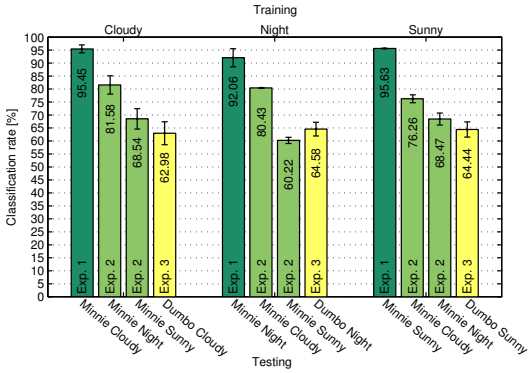
### *6.1 Stable Illumination Conditions*

In order to evaluate our method under stable illumination conditions, we trained and tested the system on pairs of image sequences taken from the IDOL database acquired one after the other using the same robot. As mentioned previously, we analyzed performance of both global (CRFH) and local (SIFT) image descriptors. We did not use the INDECS database for these experiments since only one set of data for each illumination setting was available. Although the illumination conditions for both training and test images were in this case very similar, the algorithm had to tackle other kinds of variability such as viewpoint changes (caused mainly by the manual control of the robot) and presence/absence of people. The results of the performed experiments are presented in Fig. 6a,c for CRFH and in Fig. 6b,d for SIFT. For each platform and type of illumination conditions used for training, the first bar presents an average classification rate over the two possible permutations of the image sequences in the training and test sets [1]. On average, the system classified

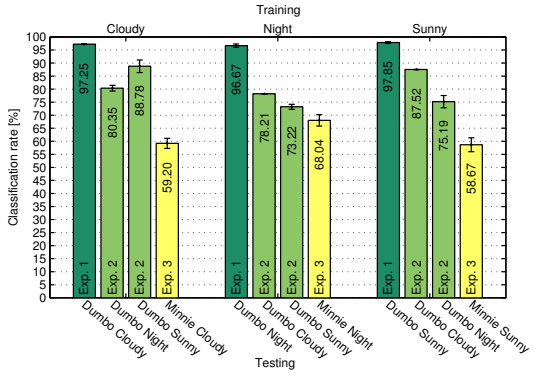
[1] Training on the first sequence, testing on the second sequence, and vice versa.



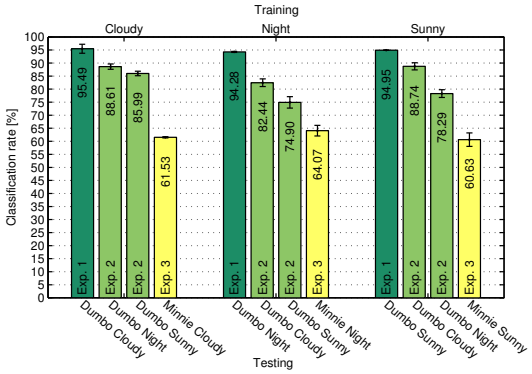

(a) Training on global features (*CRFH*) extracted from images acquired with *Minnie*.



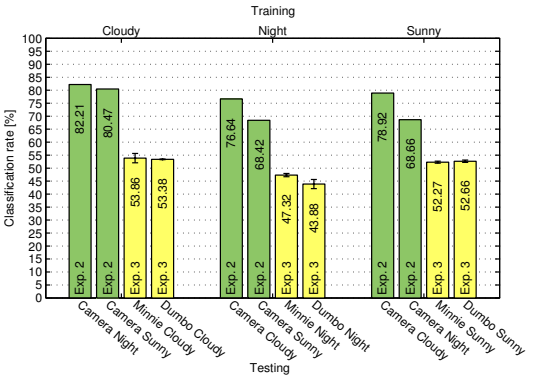

(b) Training on local features (*SIFT*) extracted from images acquired with *Minnie*.



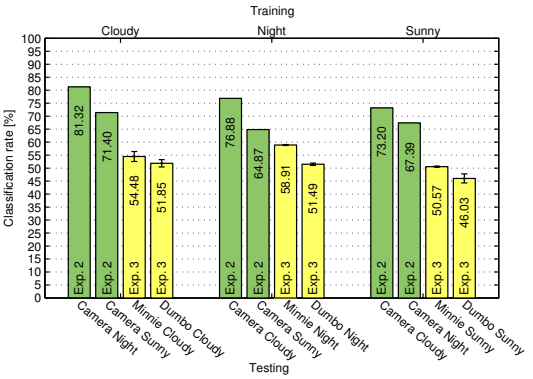

(c) Training on global features (*CRFH*) extracted from images acquired with *Dumbo*.




(d) Training on local features (*SIFT*) extracted from images acquired with *Dumbo*.




(e) Training on global features (*CRFH*) extracted from images acquired with the *standard camera*.




(f) Training on local features (*SIFT*) extracted from images acquired with the *standard camera*.

Fig. 6. Average results of the first three experiments on the IDOL and INDECS databases with both image representations. In each figure, the results are grouped according to the type of illumination conditions under which the training images were acquired. The bottom axes indicate the platform and illumination conditions used for testing. The uncertainties are given as one standard deviation.

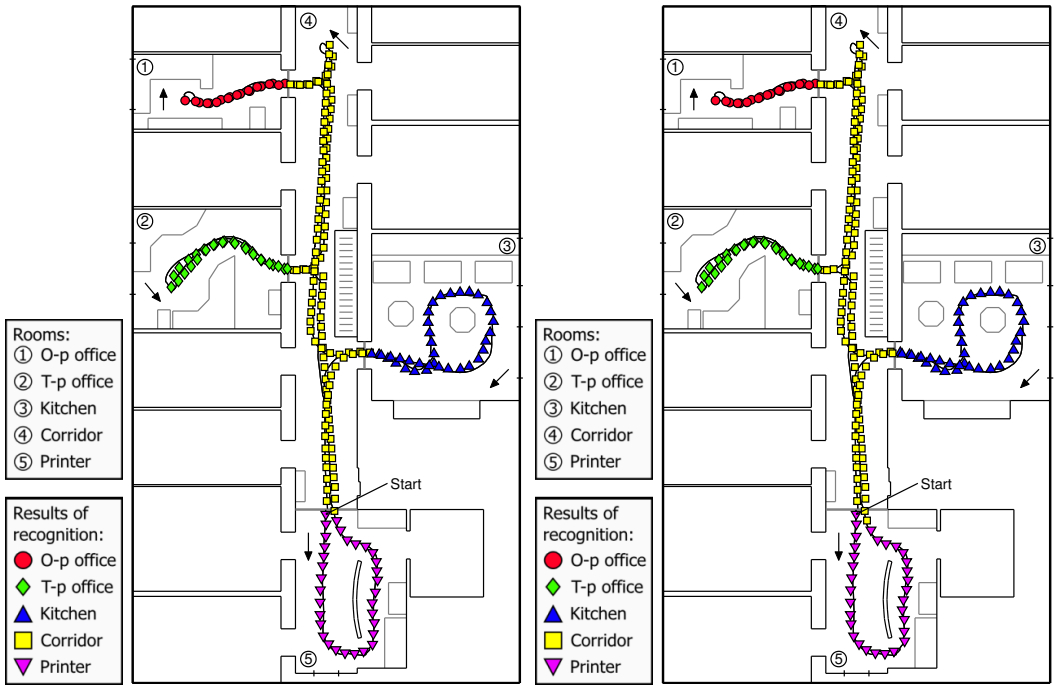
properly 95.5% of the images acquired with Minnie and 97.3% of images acquired with Dumbo when global features were used. When local features were applied, the average recognition rates were slightly lower and equal to 94.4% and 94.9% respectively.

Detailed results for two experiments conducted on data captured with each of the platforms are shown in Fig. 7. The figure presents maps of the environment with plotted paths of the robot during acquisition of the training and test sequences used during each of the experiments. Moreover, the symbols used to draw the test path indicate the results of recognition performed using image acquired at each location. Each experiment started at the point marked with the label "Start" and the arrows show the direction of driving. The position of the furniture (plotted with gray line) is approximate and sometimes slightly varied between the experiments. It can be observed that the errors are usually not a result of viewpoint variations (compare the training and test paths in the kitchen, especially in Fig. 7c,d) and mostly occur near the borders of the rooms. This can be explained by the relatively narrow field of view of the cameras as well as the fact that the images were not labeled according to their content but to the position of the robot at the time of acquisition. Since these experiments were conducted with the sequences captured under similar conditions, we treat them as a reference for other results.

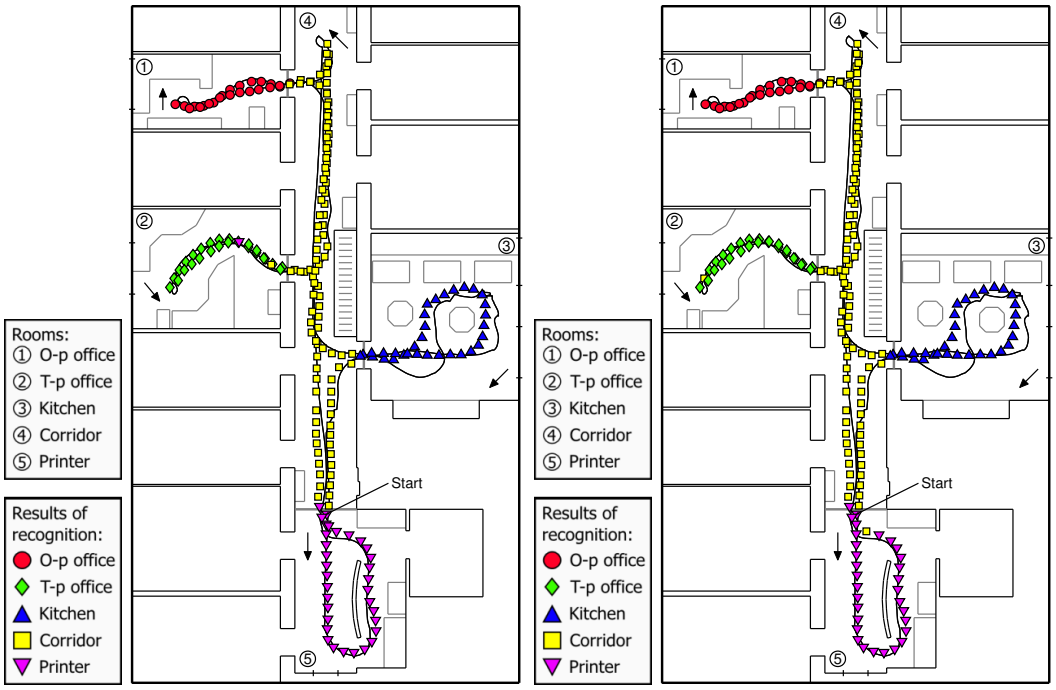
### *6.2 Varying Illumination Conditions*

We also conducted a series of experiments aiming to test the robustness of our method to changing illumination conditions as well as to other variations caused by normal activities in the rooms. The experiments were conducted on both INDECS and IDOL databases. As with the previous experiments, the same device was used for both training and testing. This time, however, the selected training and testing data sets consisted of images acquired under different illumination conditions and usually on different days. Fig. 6a-d show average results of the experiments with the image sequences from the IDOL database acquired with both robots for each permutation of the illumination conditions used for training and testing and both image representations (the two middle bars for each figure and type of training conditions). The presented classification rates obtained on the IDOL database were always averaged over two experiments with different image sequences. Fig. 6e,f gives corresponding results obtained on the INDECS database.

We see that, in general, the system performs best when trained on the images acquired in cloudy weather. The explanation for this is straightforward: the illumination conditions on a cloudy day can be seen as intermediate between those at night (only artificial light) and on a sunny day (direct natural light




(a) Dumbo Night1 ⇒ Dumbo Night2 (CRFH) (b) Dumbo Night1 ⇒ Dumbo Night2 (SIFT)




(c) Minnie Cloudy2 ⇒ Minnie Cloudy1 (CRFH) (d) Minnie Cloudy2 ⇒ Minnie Cloudy1 (SIFT)

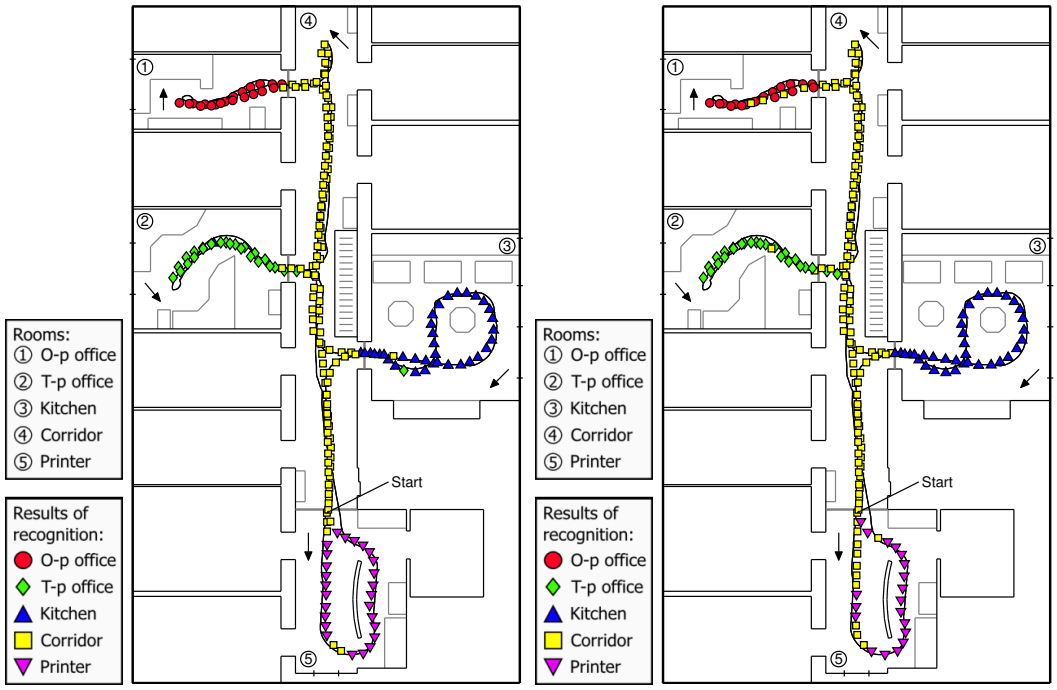
Fig. 7. Maps of the environment with plotted paths of the robot during acquisition of the training (black line) and test (points) sequences taken from the IDOL database and used during the experiments with *stable illumination conditions*. The shape of each point on the test path indicates the result of recognition.

dominates). In such case, the average classification rate computed over two testing illumination conditions (sunny and night) for both CRFH and SIFT was equal to 84.6% and 87.3% for Dumbo, 74.5% and 75.1% for Minnie, and 81.3% and 76.4% for the INDECS database. In general, local features performed slightly better than the global features (in average 71.9% vs. 72.6% for Minnie and 80.5% vs. 83.2% for Dumbo), although it was usually not the case for the INDECS database (in average 75.9% vs. 72.5%). Fig. 8 presents detailed results for two example runs and both feature types. The errors occurred mainly for the same reasons as in the previous experiments and additionally in places heavily affected by the natural light e.g. when the camera was directed towards a bright window or, in particular, large glass door in the printer area. In such cases, the automatic exposure system with which all the cameras were equipped caused the pictures to darken. Minnie was more susceptible to this phenomenon due to the higher position of its camera.

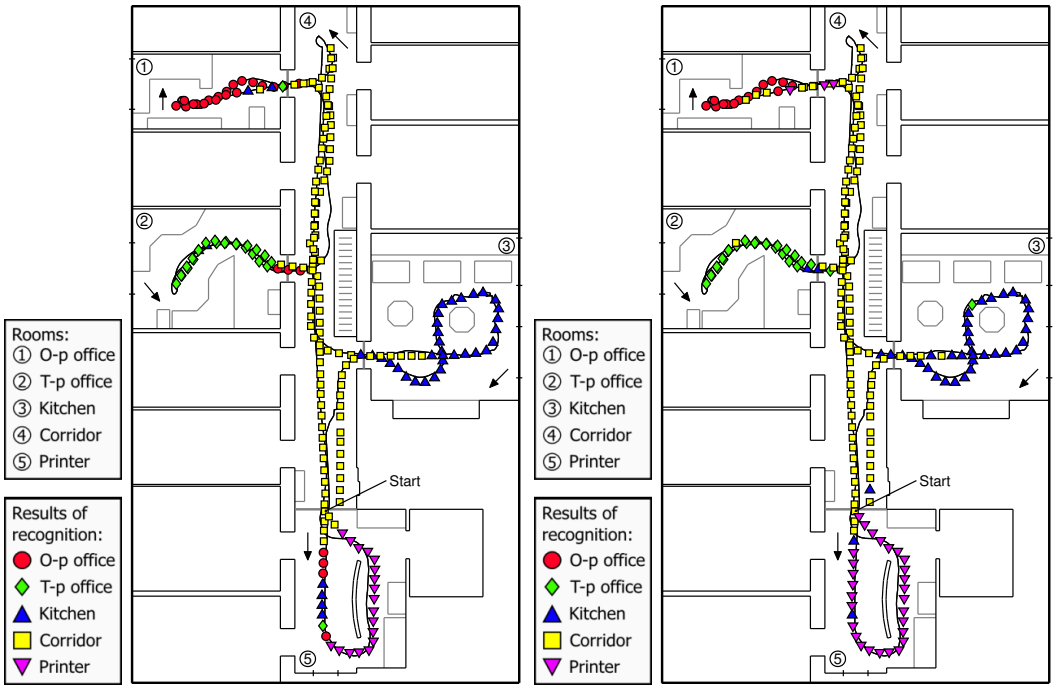
### *6.3 Recognition Across Platforms*

The third set of experiments was designed to test the portability of the acquired model across different platforms. For that purpose we trained and tested the system on image sets acquired under similar illumination conditions using different devices. We started with the experiments on image sequences from the IDOL database. We trained the system on the images acquired using either Minnie or Dumbo and tested with the images captured with the other robot. We conducted the experiments for all illumination conditions and both image representations. The main difference between the platforms from the point of view of our experiments lies in the height at which the cameras are mounted (98cm for Minnie and 36cm for Dumbo). The results presented in Fig. 6a-d indicate that our method was still able to classify correctly up to about 70% of images for CRFH and 65% of images for SIFT. There was no clear advantage of using one particular feature type. The system performed better when trained on the images captured with Minnie. This can be explained by the fact that the lower mounted camera on Dumbo provided less diagnostic information. It can also be observed from Fig. 9 that, in general, the additional errors occurred when the robot was positioned close to the walls or furniture. In such cases the height of the camera influenced the content of the images the most.

We followed a similar procedure using the INDECS database as a source of training data and different image sequences taken from the IDOL database for testing. It is important to note that the acquisition procedure differed in case of both databases, and the INDECS database was gathered ten months before the acquisition of IDOL. The points at which the pictures were taken were positioned approximately 1m from each other and, in case of the kitchen,

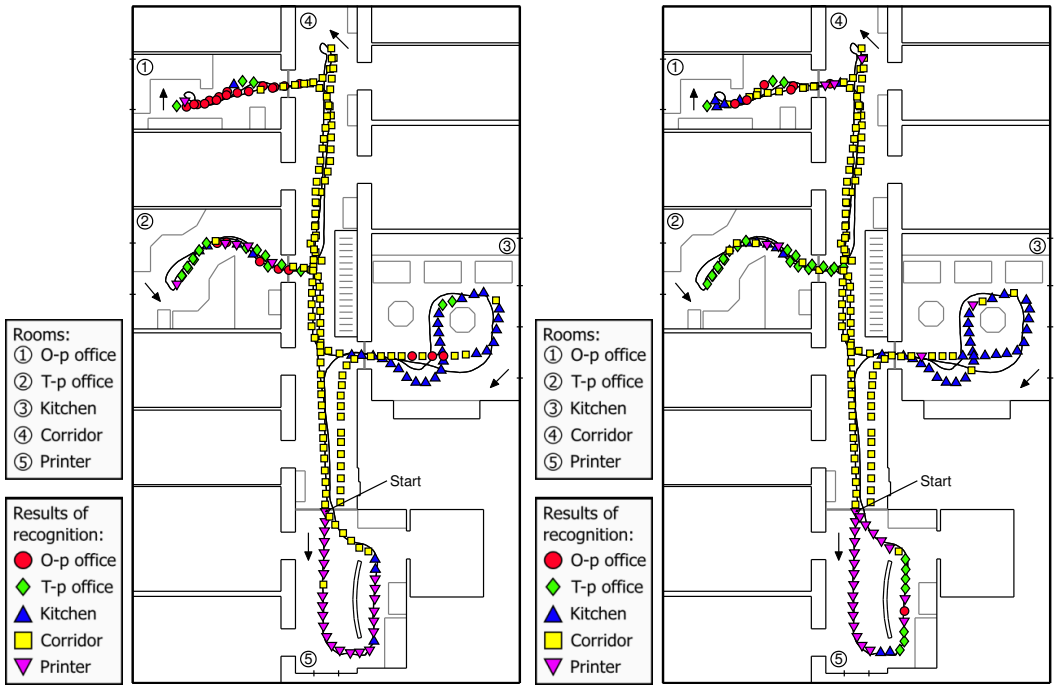



(a) Dumbo Cloudy2 ⇒ Dumbo Sunny2 (CRFH) (b) Dumbo Cloudy2 ⇒ Dumbo Sunny2 (SIFT)

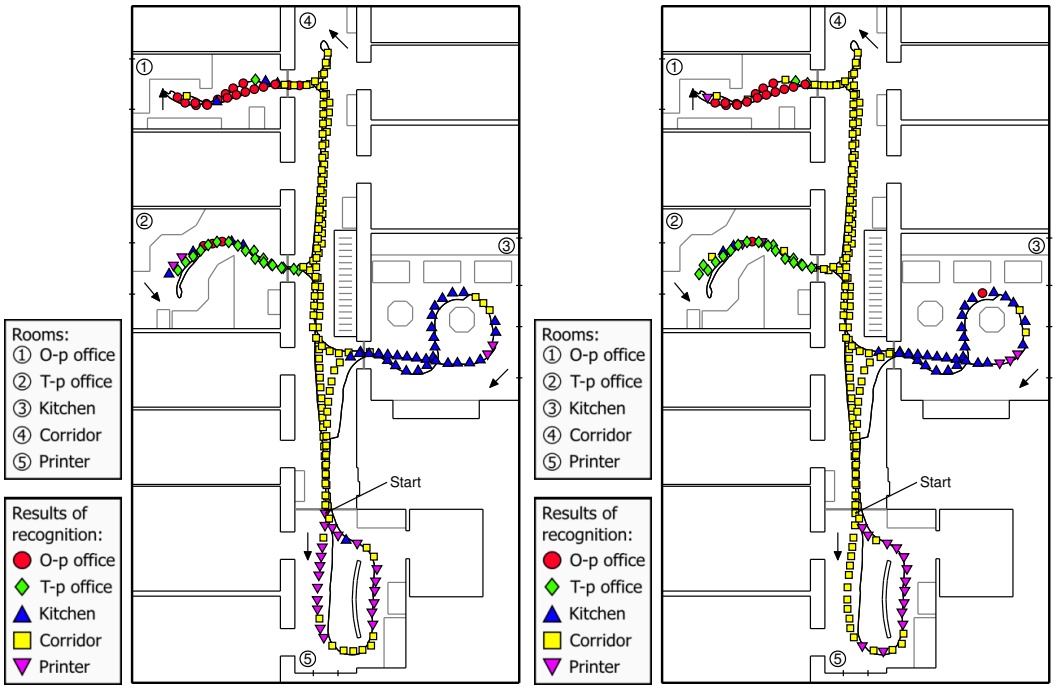



(c) Minnie Cloudy2 ⇒ Minnie Night2 (CRFH) (d) Minnie Cloudy2 ⇒ Minnie Night2 (SIFT)

Fig. 8. Maps of the environment with plotted paths of the robot during acquisition of the training (black line) and test (points) sequences taken from the IDOL database and used during the experiments with *varying illumination conditions*. The shape of each point on the test path indicates the result of recognition.

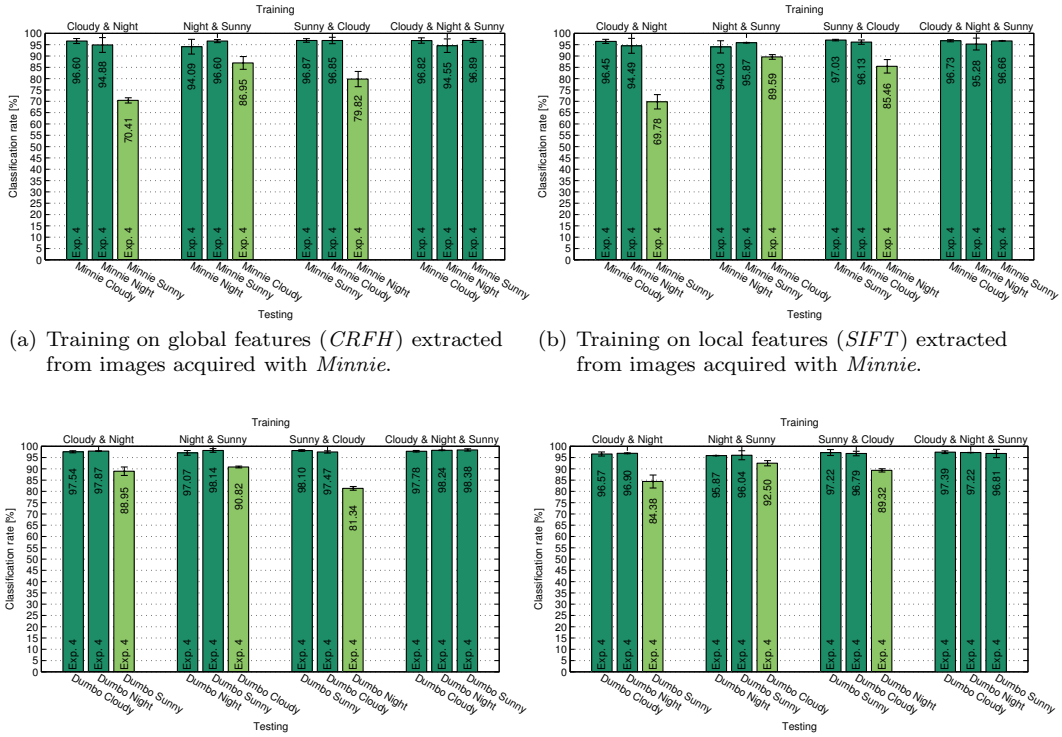



(a) Dumbo Night1 ⇒ Minnie Night2 (CRFH) (b) Dumbo Night1 ⇒ Minnie Night2 (SIFT)




(c) Minnie Sunny2 ⇒ Dumbo Sunny1 (CRFH) (d) Minnie Sunny2 ⇒ Dumbo Sunny1 (SIFT)

Fig. 9. Maps of the environment with plotted paths of the robot during acquisition of the training (black line) and test (points) sequences taken from the IDOL database and used during the experiments with *recognition across platforms*. The shape of each point on the test path indicates the result of recognition.




(a) Training on global features (*CRFH*) extracted from images acquired with *Minnie*.

(b) Training on local features (*SIFT*) extracted from images acquired with *Minnie*.

(c) Training on global features (*CRFH*) extracted from images acquired with *Dumbo*.

(d) Training on local features (*SIFT*) extracted from images acquired with *Dumbo*.

Fig. 10. Performance of the system trained on two or three image sequences acquired under different illumination conditions for both mobile platforms and image representations. The classification rates were averaged over all possible combinations of training and test sequences. The uncertainties are given as one standard deviation.

covered different area of the room due to reorganization of the furniture. Consequently, the problem required that the algorithm was invariant not only to various acquisition techniques but also offered great robustness to large changes in viewpoint and the appearance of the rooms introduced by long-time human activity. The experimental results are presented in Fig. 6e,f. We see that the algorithm obtains a recognition performance of about 50%. While this result is surely disappointing if compared to the 70% reported above, obtained for the two robot platforms, it is still quite remarkable considering the very high degree of variability between training and test data, and that results are significantly above chance (which in this case would be 20% as the datasets contain images acquired in 5 rooms).

### *6.4 Training-based Robustness*

The final series of experiments aimed at revealing whether the robustness of the recognition algorithm can be boosted by providing additional training data capturing a wider spectrum of visual variability that might occur in a real-world environment. In particular, we concentrated on invariance to

changing illumination conditions as this is the kind of variability that a continuously running visual recognition system has to deal with every day. To achieve that, we trained the system on two or three image sequences from the IDOL database gathered under different illumination conditions, and we evaluated the recognition performance on another, fourth, image set. The obtained results for both platforms, all combinations of image sequences used for training as well as both CRFH and SIFT are presented in Fig. 10a-d. The darker bars indicate the results of experiments corresponding to those discussed in Section 6.1, when training was done on an image sequence acquired under conditions similar to those used for testing. The results shown using the brighter bars can be compared with those of the experiments under varying illumination conditions analyzed in Section 6.2.

It is apparent that including images acquired under different conditions into the training set improves recognition accuracy. Although the algorithm has to incorporate much more information about each of the places into the model, the recognition accuracy for test sets acquired under similar conditions as those used for training is even greater than this obtained when each training sequence was used separately (as for the experiments discussed in Section 6.1). For example, the average recognition rate over all test sets and illumination settings for models trained on three sequences acquired using Dumbo was equal to 98.1% for CRFH and 97.1% for SIFT. At the same time, for the experiments with stable illumination conditions reported in Section 6.1 (see Fig. 6), we got only 97.3% and 94.9%. The same trend can be observed for sequences captured using Minnie. Concluding, the ability of the algorithm to handle large within-class variability is clearly not a limiting factor. It is important to note, that the recognition rate for conditions which were not used during training is also greatly improved when more training data are provided. For example, if the system was trained using the images captured during sunny weather and at night using Minnie, the average classification rate for testing image sequence acquired with cloudy weather was equal to 86.95% for CRFH and 89.59% for SIFT. Consequently, the classification rate improved by 9.9% in case of CRFH and 11.2% in case of SIFT for testing conditions not known during training, at the same time slightly improving the rates for testing conditions used also for training.

It has to be pointed out that due to the larger number of training images capturing different types of variability, the number of support vectors stored in the final model grows as well. In such case, the user pays the price of the recognition time and the memory requirements, which in case of SVMs grow linearly with the number of support vectors.

### *6.5 Discussion*

The results of the extensive experimental evaluation presented in this section indicate that our method is able to perform place recognition using standard visual sensors with high precision. It offers good robustness to changes in the illumination conditions as well as to additional variations introduced by the natural variability that occurs in real-world environments. At the same time, there is a difference in performance of the system between the experiments under stable and varying conditions, indicating that there is room for improvement in this matter.

As the system is to be used on a robot platform, it must not only be accurate but also efficient. For this reason we tried to provide the highest possible robustness using relatively small amount of training data acquired during only one run. We managed to achieve a recognition time of less than 200ms per frame on a Pentium IV 2.6 GHz using the global image representation. The results reported in Section 6.4 indicate that it is possible to significantly improve the robustness by incorporating images acquired during two or three runs under different illumination conditions into one training set. However, the higher performance does not come without a price. Since the number of support vectors in such case even doubles, the recognition time increased by about 50ms.

In all the experiments, we evaluated both global (CRFH) and local (SIFT) image descriptors. In general, we did not find any clear advantage of using one feature type over the other, and each representation has its strengths and weaknesses. The global features, however, clearly outperform SIFT in terms of efficiency, since the matching process required in order to compare two sets of local patches is computationally expensive. The efficiency of the solution based on local features could be improved by applying a more efficient matching algorithm (e.g. by using a pyramid match SVM kernel [24]) or faster interest point detector and more compact descriptor (e.g. SURF [8,39]). Since global and local representations capture different aspects of a scene, the robustness of the final solution can be further improved by integrating both cues as proposed in [48,50].

## 7 Summary

This paper discussed the need for standard benchmarking solutions for vision-based topological localization, with particular emphasis on visual place recognition. We defined and analyzed carefully the problem, and we specified the open challenges that need to be addressed by a realistic benchmark. We pre-

sented two new databases, acquired on the basis of this analysis. The first, the INDECS database, contains pictures captured with a standard camera mounted on a tripod. The second, the IDOL database, contains image sequences acquired using cameras mounted on two mobile robot platforms. The two databases were recorded within the same indoor office environment. They capture a wide spectrum of natural variations introduced by both changing illumination and human activity. Each database can be seen as a different approach to the problem; thus, they can be used to analyze different properties of a place recognition system.

We assessed both databases with a large set of baseline experiments, using a fully supervised visual place recognition system. The method employs a large-margin discriminative classifier and two different image representations: a local representation, based on SIFT features, and a global representation, consisting of multidimensional histograms of receptive fields. We conducted the experiments according to an experimental procedure designed to contain problems of varying complexity and exploit most of the variability captured in the datasets. The experimental procedure can be seen as a part of the benchmark proposed in this paper. We started from experiments performed under stable illumination settings. We then performed experiments testing the robustness of the algorithms to changing illumination and human activity. Finally, we conducted experiments with large viewpoint variations and different acquisition methods.

The reported results show that the method is able to recognize places with high precision when training and testing is performed within a relatively stable environment, or when enough training data is provided. At the same time, there is space for improvement in the robustness to illumination and large viewpoint variations. The database still poses a challenge to the system which should provide stable performance in presence of variability usually observed in real-world environments.

Finally, the dependency between the overall performance of the system and the particular set of data becomes visible as the complexity of the problem grows. Moreover, different methods (in this case different image descriptors) perform differently for different types of variations. This emphasizes the need for an extensive experimental evaluation, on a common benchmark dataset, for comparison of different approaches. When realistic datasets are available, more extensive evaluation can be conducted as the data can be reused, fully exploited, and less effort is required for acquisition and annotation. The authors believe that benchmarking solutions, such as the one presented in this paper, will make an impact on the research on visual place recognition and topological localization as was the case for other localization and visual recognition problems.

## Acknowledgment

This work was sponsored by the SSF through its Centre for Autonomous Systems (CAS), the EU integrated projects CoSy FP6-004250-IP, CogX ICT-215181 and DIRAC IST-027787 and the Swedish Research Council contract 2005-3600-Complex. The support is gratefully acknowledged.

## References

[1] ImageCLEF 2009 Robot Vision Challenge. URL: `http://imageclef.org/2009/robot/`.

[2] The KTH-TIPS image database. Available at: `http://www.nada.kth.se/cvap/databases/kth-tips/`.

[3] The MIT-CSAIL database of objects and scenes. Available at: `http://web.mit.edu/torralba/www/database.html`.

[4] The PASCAL Visual Object Classes challenge. Available at: `http://www.pascal-network.org/challenges/VOC/`.

[5] The Semantic Robot Vision Challenge. URL: `http://www.cs.cmu.edu/~srvc/`.

[6] H. Andreasson, A. Treptow, and T. Duckett. Localization for mobile robots using panoramic vision, local features and particle filter. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA'05)*, Barcelona, Spain, 2005.

[7] A. Barla, F. Odone, and A. Verri. Histogram intersection kernel for image classification. In *Proceedings of the International Conference on Image Processing (ICIP'03)*, pages 513–516, Barcelona, Spain, 2003.

[8] H. Bay, T. Tuytelaars, and L. Van Gool. SURF: Speeded up robust features. In *Proceedings of the 9th European Conference on Computer Vision (ECCV'06)*, Graz, Austria, 2006.

[9] P. Blaer and P. Allen. Topological mobile robot localization using fast vision techniques. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA'02)*, Washington, DC, USA, 2002.

[10] S. Boughorbel, J.-P. Tarel, and F. Fleuret. Non-mercer kernels for svm object recognition. In *Proceedings of the 15th British Machine Vision Conference (BMVC'04)*, London, England, September 2004.

[11] D. M. Bradley, R. Patel, N. Vandapel, and S. M. Thayer. Real-time image-based topological localization in large outdoor environments. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'05)*, Edmonton, Alberta, Canada, August 2005.

[12] E. Brunskill, T. Kollar, and N. Roy. Topological mapping using spectral clustering and classification. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'07)*, San Diego, October 2007.

[13] B. Caputo, E. Hayman, and P. Mallikarjuna. Class-specific material categorisation. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV'05)*, Beijing, China, 2005.

[14] C.-C. Chang and C.-J. Lin. *LIBSVM: a library for support vector machines*, 2001. Software available at: `http://www.csie.ntu.edu.tw/~cjlin/libsvm`.

[15] O. Chapelle, P. Haffner, and V. Vapnik. Support vector machines for histogram-based image classification. *IEEE Transactions on Neural Networks*, 10(5), May 1999.

[16] N. Cristianini and J. Shawe-Taylor. *An Introduction to Support Vector Machines and Other Kernel-based Learning Methods*. Cambridge University Press, 2000.

[17] M. Cummins and P. Newman. FAB-MAP: Probabilistic localization and mapping in the space of appearance. *The International Journal of Robotics Research*, 27(6):647–665, June 2008.

[18] M. Dissanayake, P. Newman, S. Clark, H. Durrant-Whyte, and M. Csorba. A solution to the simultaneous localization and map building (SLAM) problem. *IEEE Transactions on Robotics and Automation*, 17(3):229–241, 2001.

[19] G. Dorkó and C. Schmid. Object class recognition using discriminative local features. 2005.

[20] D. Filliat. A visual bag of words method for interactive qualitative localization and mapping. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA'07)*, Roma, Italy, April 2007.

[21] J. Folkesson, P. Jensfelt, and H. Christensen. Vision SLAM in the measurement subspace. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA'05)*, pages 30–35, Barcelona, Spain, 2005.

[22] F. Fraundorfer, C. Engels, and D. Nistér. Topological mapping, localization and navigation using image collections. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'07)*, San Diego, CA, USA, October 2007.

[23] J. Gaspar, N. Winters, and J. Santos-Victor. Vision-based navigation and environmental representations with an omni-directional camera. *IEEE Transactions on Robotics and Automation*, 16(6), 2000.

[24] K. Grauman and T. Darrell. The pyramid match kernel: Discriminative classification with sets of image features. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV'05)*, Beijing, China, October 2005.

[25] G. Griffin, A. Holub, and P. Perona. Caltech-256 Object Category Dataset. Technical Report 7694, Caltech, 2007. Available at: `http://authors.library.caltech.edu/7694/`.

[26] A. Howard and N. Roy. The Robotics Data Set Repository (Radish), 2003. Available at: `http://radish.sourceforge.net/`.

[27] M. Jogan and A. Leonardis. Robust localization using an omnidirectional appearance-based subspace model of environment. *Robotics and Autonomous Systems*, 45(1):51–72, October 2003.

[28] D. Kortenkamp and T. Weymouth. Topological mapping for mobile robots using a combination of sonar and vision sensing. In *Proceedings of the 12th National Conference on Artificial Intelligence (AAAI'94)*, Seattle, Washington, USA, 1994.

[29] B. Kuipers and P. Beeson. Bootstrap learning for place recognition. In *Proceedings of the 18th National Conference on Artificial Intelligence (AAAI'02)*, 2002.

[30] O. Linde and T. Lindeberg. Object recognition using composed receptive field histograms of higher dimensionality. In *Proceedings of the 17th International Conference on Pattern Recognition (ICPR'04)*, Cambridge, UK, 2004.

[31] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004.

[32] J. Luo, A. Pronobis, B. Caputo, and P. Jensfelt. The KTH-IDOL2 database. Technical Report CVAP304, Kungliga Tekniska Högskolan, CVAP/CAS, October 2006. Available at `http://cogvis.nada.kth.se/IDOL/`.

[33] M. Marszałek, I. Laptev, and C. Schmid. Actions in context. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'09)*, 2009. Available at: `http://www.irisa.fr/vista/actions/hollywood2/`.

[34] M. Mata, J. M. Armingol, A. de la Escalera, and S. M. A. Using learned visual landmarks for intelligent topological navigation of mobile robots. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA'03)*, 2003, Taipei, Taiwan.

[35] E. Menegatti, M. Zoccarato, E. Pagello, and H. Ishiguro. Image-based monte-carlo localisation with omnidirectional images. *Robotics and Autonomous Systems*, 48(1), 2004.

[36] K. Mikolajczyk and C. Schmid. Indexing based on scale invariant interest points. In *Proceedings of the 8th IEEE International Conference on Computer Vision (ICCV'01)*, Vancouver, Canada, 2001.

[37] K. Mikolajczyk and C. Schmid. A performance evaluation of local descriptors. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'03)*, Madison, WI, USA, 2003.

[38] O. Mozos, R. Triebel, P. Jensfelt, A. Rottmann, and W. Burgard. Supervised semantic labeling of places using information extracted from sensor data. *Robotics and Autonomous Systems*, 55(5), 2007.

[39] A. C. Murillo, J. J. Guerrero, and C. Sagues. SURF features for efficient robot localization with omnidirectional images. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA'07)*, Roma, Italy, April 2007.

[40] E. Nebot. The Sydney Victoria Park dataset. Available at: `http://www-personal.acfr.usyd.edu.au/nebot/dataset.htm`.

[41] M. E. Nilsback and B. Caputo. Cue integration through discriminative accumulation. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'04)*, Washington, DC, USA, 2004.

[42] I. Nourbakhsh, R. Powers, and S. Birchfield. Dervish: An office navigation robot. *AI Magazine*, 16(2):53–60, 1995.

[43] J. C. Platt. Fast training of support vector machines using sequential minimal optimization. In B. Schölkopf, C. Burges, and A. Smola, editors, *Advances in Kernel Methods: Support Vector Learning*, pages 185–208. MIT Press, Cambridge, MA, USA, 1999.

[44] J. C. Platt, N. Cristianini, and J. Shawe-Taylor. Large margin DAGs for multiclass classification. In *Advances in Neural Information Processing Systems*, volume 12, pages 547–553, 2000.

[45] J. Ponce, T. L. Berg, M. Everingham, D. Forsyth, M. Hebert, S. Lazebnik, M. Marszałek, C. Schmid, C. Russell, A. Torralba, C. Williams, J. Zhang, and A. Zisserman. Dataset issues in object recognition. In *Towards Category-Level Object Recognition*, pages 29–48. Springer, 2006.

[46] A. Pronobis. Indoor place recognition using support vector machines. Master's thesis, NADA/CVAP, Kungliga Tekniska Högskolan, Stockholm, Sweden, December 2005. Available at `http://www.csc.kth.se/~pronobis/`.

[47] A. Pronobis and B. Caputo. The KTH-INDECS database. Technical Report CVAP297, Kungliga Tekniska Hoegskolan, CVAP, September 2005. Available at `http://cogvis.nada.kth.se/INDECS/`.

[48] A. Pronobis and B. Caputo. Confidence-based cue integration for visual place recognition. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'07)*, San Diego, CA, USA, October 2007.

[49] A. Pronobis, B. Caputo, P. Jensfelt, and H. I. Christensen. A discriminative approach to robust visual place recognition. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'06)*, Beijing, China, October 2006.

[50] A. Pronobis, O. Mozos, and B. Caputo. SVM-based discriminative accumulation scheme for place recognition. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA'08)*, Pasadena, CA, USA, May 2008.

[51] C. Siagian and L. Itti. Biologically-inspired robotics vision monte-carlo localization in the outdoor environment. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'07)*, San Diego, CA, USA, October 2007.

[52] H. Tamimi and A. Zell. Vision based localization of mobile robots using kernel approaches. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'04)*, Sendai, Japan, 2004.

[53] A. Tapus and R. Siegwart. Incremental robot mapping with fingerprints of places. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'05)*, Edmonton, Alberta, Canada, August 2005.

[54] S. Thrun. Learning metric-topological maps for indoor mobile robot navigation. *Artificial Intelligence*, 1999(1), 1998.

[55] A. Torralba. Contextual priming for object detection. *International Journal of Computer Vision*, 53(2), 2003.

[56] A. Torralba, K. P. Murphy, W. T. Freeman, and M. A. Rubin. Context-based vision system for place and object recognition. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV'03)*, 2003, Nice, France.

[57] A. Torralba and P. Sinha. Recognizing indoor scenes. Technical Report 2001-015, AI Memo, 2001.

[58] M. M. Ullah, A. Pronobis, B. Caputo, J. Luo, P. Jensfelt, and H. I. Christensen. Towards robust place recognition for robot localization. In *Proceedings of the 2008 IEEE International Conference on Robotics and Automation (ICRA'08)*, Pasadena, CA, USA, May 2008.

[59] I. Ulrich and I. Nourbakhsh. Appearance-based place recognition for topological localization. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA'00)*, San Francisco, CA, USA, 2000.

[60] C. Valgren and A. J. Lilienthal. Incremental spectral clustering and seasons: Appearance-based localization in outdoor environments. In *Proceedings of the International Conference on Robotics and Automation (ICRA'08)*, 2008.

[61] C. Wallraven, B. Caputo, and A. Graf. Recognition with local features: the kernel recipe. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV'03)*, Nice, France, 2003.

[62] C. Weiss, H. Tamimi, A. Masselli, and A. Zell. A hybrid approach for vision-based outdoor robot localization using global and local image features. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'07)*, San Diego, CA, USA, October 2007.

[63] J. Wolf, W. Burgard, and H. Burkhardt. Robust vision-based localization by combining an image retrieval system with monte carlo localization. *IEEE Transactions on Robotics*, 21(2):208–216, 2005.

[64] H. Zender, O. M. Mozos, P. Jensfelt, G.-J. M. Kruijff, and W. Burgard. Conceptual spatial representations for indoor mobile robots. *Robotics and Autonomous Systems*, 56(6):493–502, June 2008.

## Multi-modal Semantic Place Classification


A. Pronobis, O. Martínez Mozos, B. Caputo and P. Jensfelt



*The International Journal of Robotics Research* 2010; 29; 298 originally published online Dec 4, 2009;
DOI: 10.1177/0278364909356483

# Multi-modal Semantic Place Classification


## A. Pronobis

Centre for Autonomous Systems,
Royal Institute of Technology,
SE-100 44 Stockholm, Sweden
pronobis@csc.kth.se

## O. Martínez Mozos

Department of Computer Science,
University of Freiburg,
D-79110, Freiburg, Germany
omartine@informatik.uni-freiburg.de

## B. Caputo

Idiap Research Institute,
CH-1920 Martigny, Switzerland
bcaputo@idiap.ch

## P. Jensfelt

Centre for Autonomous Systems,
Royal Institute of Technology,
SE-100 44 Stockholm, Sweden
patric@csc.kth.se



## Abstract

*The ability to represent knowledge about space and its position therein is crucial for a mobile robot. To this end, topological and semantic descriptions are gaining popularity for augmenting purely metric space representations. In this paper we present a multi-modal place classification system that allows a mobile robot to identify places and recognize semantic categories in an indoor environment. The system effectively utilizes information from different robotic sensors by fusing multiple visual cues and laser range data. This is achieved using a high-level cue integration scheme based on a Support Vector Machine (SVM) that learns how to optimally combine and weight each cue. Our multi-modal place classification approach can be used to obtain a real-time semantic space labeling system which integrates information over time and space. We perform an extensive experimental evaluation of the method for two different platforms and environments, on a realistic off-line database and in a live experiment on an autonomous robot. The results clearly demonstrate the effectiveness of our cue integration scheme and its value for robust place classification under varying conditions.*



KEY WORDS—recognition, sensor fusion, localization, multi-modal place classification, sensor and cue integration, semantic annotation of space


## 1. Introduction

The most fundamental competence for an autonomous mobile agent is to know its position in the world. This can be represented in terms of raw metric coordinates, topological location, or even semantic description. Recently, there has been a growing interest in augmenting (or even replacing) purely metric space representations with topological and semantic place information. Several attempts have been made to build autonomous cognitive agents able to perform human-like tasks[1]. Enhancing the space representation to be more meaningful from the point of view of spatial reasoning and human–robot interaction have been at the forefront of the issues being addressed (Kuipers 2006; Topp and Christensen 2006; Zender et


The International Journal of Robotics Research
Vol. 29, No. 2–3, February/March 2010, pp. 298–320
DOI: 10.1177/0278364909356483



Figures 1–15, 17, 18 appear in color online: http://ijr.sagepub.com


1. See, e.g., CoSy (Cognitive Systems for Cognitive Assistants) http://www.cognitivesystems.org/ or COGNIRON (the cognitive robot companion) http://www.cogniron.org.

al. 2008). Indeed, in the concrete case of indoor environments, the ability to understand the existing topological relations and associate semantic terms such as "corridor" or "office" with places, gives a much more intuitive idea of the position of the robot than global metric coordinates. In addition, the semantic information about places can extend the capabilities of a robot in other tasks such as localization (Rottmann et al. 2005), exploration (Stachniss et al. 2006), or navigation (Galindo et al. 2005).

Nowadays, robots are usually equipped with several sensors providing both geometrical and visual information about the environment. Previous work on place classification relied on sonar and/or laser range data as robust sensory modalities (Mozos et al. 2005). However, the advantages of geometric solutions, such as invariance to visual variations and low dimensionality of the processed information, quickly became their weaknesses. The inability to capture many aspects of complex environments leads to the problem of perceptual aliasing (Kuipers and Beeson 2002) and can limit the usefulness of such methods for topological and semantic mapping. Recent advances in vision have made this modality emerge as a natural and viable alternative. Vision provides richer sensory input allowing for better discrimination. Moreover, a large share of the semantic description of a place is encoded in its visual appearance. However, visual information tends to be noisy and difficult to interpret as the appearance of places varies over time due to changing illumination and human activity. At the same time, the visual variability within place classes is huge, making the semantic place classification a challenging problem. Clearly, each modality has its own characteristics. Interestingly, the weaknesses of one often correspond to the strengths of the other.

In this paper, we propose an approach to semantic place classification which combines the stability of geometrical solutions with the versatility of vision. First, we present a recognition system implemented on a mobile robot platform integrating multiple cues and modalities. The system is able to perform robust place classification under different types of variations that occur in indoor environments over a span of time of several months. This comprises variations in illumination conditions and in configuration of furniture and small objects. The system relies on different types of visual information provided by global and local descriptors and on geometric cues derived from laser range scans. For the vision channel we apply the Scale-Invariant Feature Transform (SIFT) (Lowe 2004) and Composed Receptive Field Histograms (CRFH) (Linde and Lindeberg 2004). For the laser channel we use the features proposed in Mozos et al. (2005, 2007).

We combine the cues using a new high-level accumulation scheme, which builds on our previous work (Nilsback and Caputo 2004; Pronobis and Caputo 2007). We train for each cue a large margin classifier which outputs a set of scores encoding confidence of the decision. Integration is then achieved by feeding the scores to a Support Vector Machine (SVM) (Cristianini and Shawe-Taylor 2000). Such an approach allows to optimally combine cues, even obtained using different types of models, with a complex, possibly non-linear function. We call this algorithm the SVM-based Discriminative Accumulation Scheme (SVM-DAS).

Finally, we show how to build a self-contained semantic space labeling system, which relies on multi-modal place classification as one of its components. The system is implemented as a part of an integrated cognitive robotic architecture[2] and runs on-line on a mobile robot platform. While the robot explores the environment, the system acquires evidence about the semantic category of the current area produced by the place classification component and accumulates them both over time and space. As soon as the system is confident about its decision, the area is assigned a semantic label. We integrate the system with a Simultaneous Localization and Mapping (SLAM) algorithm and show how a metric and topological space representation can be augmented with a semantic description.

We evaluated the robustness of the presented methods in several sets of extensive experiments. We conducted experiments on two different robot platforms, in two different environments and for two different scenarios. First, we run a series of off-line experiments of increasing difficulty on the IDOL2 database (Luo et al. 2006) to precisely measure the performance of the place classification algorithm in presence of different types of variations. These ranged from short-term visual variations caused by changing illumination to long-term changes which occurred in the office environment over several months. Second, we run a live experiment where a robot performs SLAM and semantic labeling in a new environment using prebuilt models of place categories. Results show that integrating different visual cues, as well as different modalities, allows to greatly increase the robustness of the recognition system, achieving high accuracy under severe dynamic variations. Moreover, the place classification system, when used in the framework of semantic space labeling, can yield a fully correct semantic representation even for a new, unknown environment.

The rest of the paper is organized as follows. After a review of the related literature (Section 2), Section 3 presents the main principle behind our multi-modal place classification algorithm and describes the methods used to extract each cue. Then, Section 4 gives details about the new cue integration scheme and Section 5 describes the architecture of the semantic labeling system. Finally, Section 6 presents detailed experimental evaluation of the place classification system and Section 7 reports results of the live experiment with semantic labeling of space. The paper concludes with a summary and possible avenues for future research.

2. See CoSy (Cognitive Systems for Cognitive Assistants) http://www.cognitivesystems.org/ and CAST (The CoSy Architecture Schema Toolkit) http://www.cs.bham.ac.uk/research/projects/cosy/cast/.

## 2. Related Work

Place classification is a vastly researched topic in the robotic community. Purely geometric solutions based on laser range data have proven to be successful for certain tasks and several approaches were proposed using laser scanners as the only sensors. Koenig and Simmons (1998) used a pre-programmed routine to detect doorways from range data. In addition, Althaus and Christensen (2003) used line features to detect corridors and doorways. In their work, Buschka and Saffiotti (2002) partitioned grid maps of indoor environments into two different classes of open spaces, i.e. rooms and corridors. The division of the open spaces was done incrementally on local submaps. Finally, Mozos et al. (2005) applied boosting to create a classifier based on a set of geometrical features extracted from range data to classify different places in indoor environments. A similar idea was used by Topp and Christensen (2006) to describe regions from laser readings.

The limitations of geometric solutions inspired many researchers to turn towards vision which nowadays becomes tractable in real-time applications. The proposed methods employed either perspective (Torralba et al. 2003; Tamimi and Zell 2004; Filliat 2007) or omnidirectional cameras (Gaspar et al. 2000; Ulrich and Nourbakhsh 2000; Blaer and Allen 2002; Menegatti et al. 2004; Andreasson et al. 2005; Murillo et al. 2007; Valgren and Lilienthal 2008). The main differences between the approaches relate to the way the scene is perceived, and thus the method used to extract characteristic features from the scene. Landmark-based techniques make use of either artificial or natural landmarks in order to extract information about a place. Siagian and Itti (2007) relied on visually distinctive image regions as landmarks. Many other solutions employed local image features, with SIFT being the most frequently applied (Se et al. 2001; Lowe 2004; Andreasson et al. 2005; Pronobis and Caputo 2007). Zivkovic et al. (2005) used the SIFT descriptor to build a topological representation by clustering a graph representing relations between images. Other approaches used the bag-of-words technique (Filliat 2007; Fraundorfer et al. 2007), the SURF features (Bay et al. 2006; Murillo et al. 2007; Valgren and Lilienthal 2008), or representation based on information extracted from local patches using Kernel PCA (Tamimi and Zell 2004). Global features are also commonly used for place recognition. Torralba et al. (Torralba and Sinha 2001; Torralba et al. 2003; Torralba 2003) suggested to use a representation called the "gist" of a scene, which is a vector of principal components of outputs of a bank of spatially organized filters applied to the image. Other approaches use color histograms (Ulrich and Nourbakhsh 2000; Blaer and Allen 2002), gradient orientation histograms (Bradley et al. 2005), eigenspace representation of images (Gaspar et al. 2000), or Fourier coefficients of low-frequency image components (Menegatti et al. 2004).

In all of the previous approaches only one modality is used for the recognition of places. Recently, several authors observed that robustness and efficiency of the recognition system can be improved by combining information provided by different visual cues. Siagian and Itti (2007) and Weiss et al. (2007) used a global representation of the images together with local visual landmarks to localize a robot in outdoor environments. Pronobis and Caputo (2007) used two cues composed of global and local image features to recognize places in indoor environments. The cues were combined using discriminative accumulation. Here, we extend this approach by integrating information provided by a laser range sensor using a more sophisticated algorithm.

Other approaches also employed a combination of different sensors, mainly laser and vision. Tapus and Siegwart (2005) used an omnidirectional camera and two lasers covering 360° field of view to extract fingerprints of places for topological mapping. This approach was not used for extracting semantic information about the environment. Posner et al. (2007) and Douillard et al. (2007) relied on range data and vision for recognition of objects in outdoor environments (e.g. grass, walls, or cars). Finally, Rottmann et al. (2005) used a combination of both modalities to categorize places in indoor environments. Each observation was composed of a set of geometrical features and a set of objects found in the scene. The geometrical features were calculated from laser scans and the objects were detected using Haar-like features from images. The extracted information was integrated at the feature level. In contrast, the method presented in this work learns how to combine and weigh outputs of several classifiers, keeping features and therefore the information from different modalities separate.

Various cue integration methods have been proposed in the robotics and machine learning community (Poggio et al. 1985; Matas et al. 1995; Triesch and Eckes 1998; Nilsback and Caputo 2004; Tapus and Siegwart 2005; Pronobis and Caputo 2007). These approaches can be described according to various criteria. For instance, Clark and Yuille (1990) suggest to classify them into two main groups, *weak coupling* and *strong coupling*. Assuming that each cue is used as input of a different classifier, weak coupling is when the output of two or more independent classifiers are combined. Strong coupling is when the output of one classifier is affected by the output of another classifier, so that their outputs are no longer independent. Another possible classification is into *low-level* and *high-level* integration methods, where the emphasis is on the level at which integration happens. We call *low-level integration methods* those algorithms where cues are combined together at the feature level, and then used as input to a single classifier. This approach has been used successfully for object recognition using multiple visual cues (Matas et al. 1995), and for topological mapping using multiple sensor modalities (Tapus and Siegwart 2005). In spite of remarkable performances for specific tasks, there are several drawbacks of the low-level methods. First, if one of the cues gives misleading information, it is quite probable that the new feature vector will be adversely affected influencing the whole performance. Second, we can

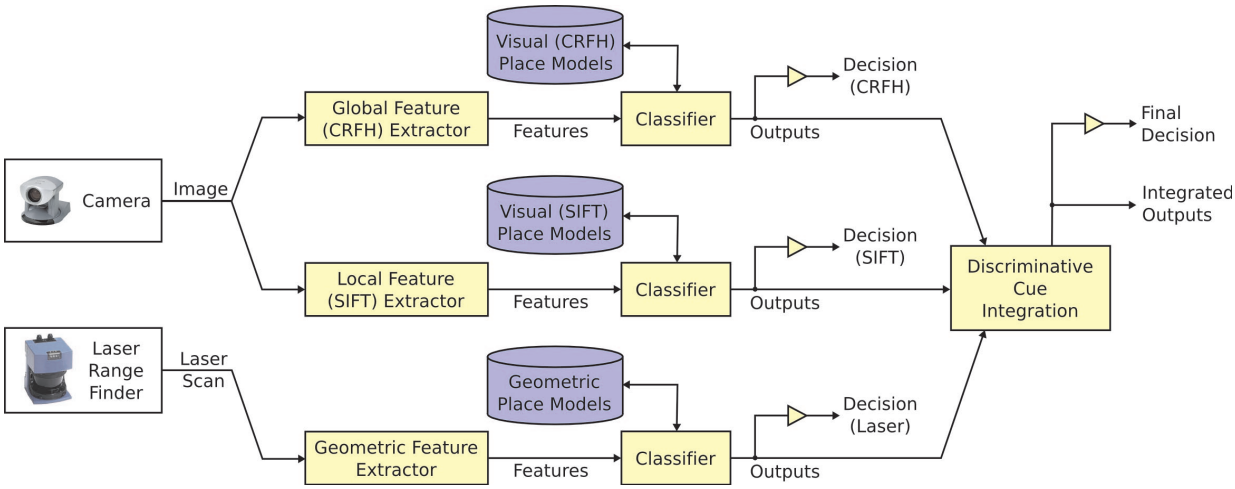



Fig. 1. Architecture of the multi-modal place classification system.

expect the dimension of such a feature vector to increase as the number of cues grow, and each of the cues needs to be used even if one would allow for correct classification. This implies longer learning and recognition times, greater memory requirements, and possible curse of dimensionality effects. Another strategy is to keep the cues separated and to integrate the outputs of individual classifiers, each trained on a different cue (Poggio et al. 1985; Nilsback and Caputo 2004; Pronobis and Caputo 2007). We call such algorithms *high-level integration methods*, of which voting is the most popular (Duda et al. 2001). These techniques are more robust with respect to noisy cues or sensory channels, allow the use of different classifiers adapted to the characteristics of each single cue and decide on the number of cues that should be extracted and used for each particular classification task (Pronobis and Caputo 2007). In this paper, we focus on a weak coupling, high-level integration method called *accumulation*. The underlying idea is that information from different cues can be summed together, thus accumulated. The idea was first proposed in probabilistic framework by Poggio et al. (1985) and further explored by Aloimonos and Shulman (1989). The method was then extended to discriminative methods in Nilsback and Caputo (2004) and Pronobis and Caputo (2007).

## 3. Multi-modal Place Classification

The ability to integrate multiple cues, possibly extracted from different sensors, is an important skill for a mobile robot. Different sensors usually capture different aspects of the environment. Therefore using multiple cues leads to obtaining a more descriptive representation. The visual sensor is an irreplaceable source of distinctive information about a place. However, this information tends to be noisy and difficult to analyze due to the susceptibility to variations introduced by changing illumination and everyday activities in the environment. At the same time, most recent robotic platforms are equipped with a laser range scanner which provides much more stable and robust geometric cues. These cues however, are unable to uniquely represent the properties of different places (perceptual aliasing) (Kuipers and Beeson 2002). Clearly performance could increase if different cues were combined effectively. Note that even alternative interpretations of the information obtained by the same sensor can be valuable, as we will show experimentally in Section 6.

This section describes our approach to multi-modal place classification. Our method is fully supervised and assumes that during training, each place (room) is represented by a collection of labeled data which captures its intrinsic visual and geometric properties under various viewpoints, at a fixed time and illumination setting. During testing, the algorithm is presented with data samples acquired in the same places, under roughly similar viewpoints but possibly under different conditions (e.g. illumination), and after some time (where the time range goes from some minutes to several months). The goal is to recognize correctly each single data sample provided to the system.

The architecture of the system is illustrated in Figure 1. We see that there is a separate path for each cue. We use two different visual cues corresponding to two types of image features (local and global) as well as simple geometrical features extracted from laser range scans. Each path consists of two main building blocks: a feature extractor and a classifier. Thus, separate decisions can be obtained for each of the cues in case only one cue is available. Alternatively, our method could decide when to acquire additional information (e.g. only in difficult cases) (Pronobis and Caputo 2007). In cases when several cues are available, the outputs encoding the confidence of the single-cue classifiers are combined using an efficient discriminative accumulation scheme.

The rest of this section gives details about the algorithms used to extract and classify each of the cues for the vision-based paths (Section 3.1) and laser-based path (Section 3.2). A comprehensive description of the algorithms used for cue integration is given in Section 4.

### *3.1. Vision-based Place Classification*

As a basis for the vision-based channel, we used the place recognition system presented in Pronobis et al. (2006) and Pronobis and Caputo (2007), which is built around a SVM classifier (Cristianini and Shawe-Taylor 2000) and two types of visual features, global and local, extracted from the same image frame. We used CRFH (Linde and Lindeberg 2004) as global features, and SIFT (Lowe 2004) as local features. Both have already been proved successful in the domain of vision-based place recognition (Pronobis et al. 2006; Pronobis and Caputo 2007) and localization and mapping (Se et al. 2001; Andreasson et al. 2005).

CRFHs are a sparse multi-dimensional statistical representation of responses of several image filters applied to the input image. Following Pronobis et al. (2006), we used histograms of six dimensions, with 28 bins per dimension, computed from second-order normalized Gaussian derivative filters applied to the illumination channel at two scales. The SIFT descriptor instead represents local image patches around interest points characterized by coordinates in the scalespace in the form of histograms of gradient directions. To find the coordinates of the interest points, we used a scale and affine invariant region detector based on the difference-of-Gaussians (DoG) operator (Rothganger et al. 2006).

We used SVMs for creating models from both visual cues. A review of the theory behind SVMs can be found in Section 4.1. In case of SVMs, special care must be taken in choosing an appropriate kernel function. Here we used the $\chi^2$ kernel (Chapelle et al. 1999) for CRFH, and the match kernel proposed by Wallraven et al. (2003) for SIFT. Both have been used in our previous work on SVM-based place recognition, obtaining good performances.

### *3.2. Laser-based Place Classification*

In addition to the visual channel, we used a laser range sensor. A single two-dimensional (2D) laser scan covered a field of view of 180° in front of the robot. A laser observation $z = \{b_0, \ldots, b_{M-1}\}$ contains a set of beams $b_i$, in which each beam $b_i$ consists of a tuple $(\alpha_i, d_i)$, where $\alpha_i$ is the angle of the beam relative to the robot and $d_i$ is the length of the beam.

For each laser observation, we calculated a set of simple geometric features represented by single real values. The features were introduced for place classification by Mozos et al. (2005) where laser observations covering a 360° field of view were used. The complete set of features consists of two subsets. The first subset contains geometrical features calculated directly from the laser beams. The second subset comprises geometrical features extracted from a polygon approximation of the laser observation. This polygon is created by connecting the end points of the beams. The selection of features is based on the results presented in Mozos et al. (2005, 2007).

As classifiers for the laser-based channel, we tried both AdaBoost (Freund and Schapire 1995), following the work in Mozos et al. (2007), and SVMs. In the rest of the paper, we will refer to the two laser-based models as L-AB and L-SVM, respectively. For the geometric features, we used a Radial Basis Function (RBF) kernel (Cristianini and Shawe-Taylor 2000) with SVMs, chosen through a set of reference experiments[3]. Both classifiers were benchmarked on the laser-based place classification task. Results presented in Section 6.2 show an advantage of the more complex SVM classifier.

## 4. Discriminative Cue Integration

This section describes our approach to cue integration from one or multiple modalities. We propose an SVM-DAS, a technique performing non-linear cue integration by discriminative accumulation. For each cue, we train a separate large margin classifier which outputs a set of scores (outputs), encoding the confidence of the decision. We achieve integration by feeding the scores to an SVM. Compared to previous accumulation methods (Poggio et al. 1985; Caputo and Dorko 2002; Nilsback and Caputo 2004; Pronobis and Caputo 2007), SVM-DAS gives several advantages: (a) discriminative accumulation schemes achieve consistently better performances than probabilistic ones (Poggio et al. 1985; Caputo and Dorko 2002), as shown in Nilsback and Caputo (2004); (b) compared with previous discriminative accumulation schemes (Nilsback and Caputo 2004; Pronobis and Caputo 2007), our approach accumulates cues with a more complex, possibly non-linear function, by using the SVM framework and kernels (Cristianini and Shawe-Taylor 2000). Such an approach makes it possible to integrate outputs of different classifiers such as SVM and AdaBoost. At the same time, it learns the weight for each cue very efficiently, therefore making it possible to accumulate large numbers of cues without computational problems.

In the rest of the section we first sketch the theory behind SVMs (Section 4.1), a crucial component in our approach. We then describe the Generalized Discriminative Accumulation Scheme (G-DAS; see Pronobis and Caputo (2007) and Section 4.2) on which to a large extent we build. Finally, we introduce the new algorithm and discuss its advantages in Section 4.3.

3. In the case of AdaBoost, we constructed a multi-class classifier by arranging several binary classifiers into a decision list in which each element corresponded to one specific class.

### *4.1. Support Vector Machines*

Consider the problem of separating the set of labeled training data $(\boldsymbol{x}_1, y_1), \ldots, (\boldsymbol{x}_n, y_n), \ldots, (\boldsymbol{x}_N, y_N)$ into two classes, where $\boldsymbol{x}_n \in \Re^L$ is a feature vector and $y_n \in \{-1, +1\}$ its class label. Assuming that the two classes can be separated by a hyperplane in some Hilbert space $\mathcal{H}$, then the optimal separating hyperplane is the one which has maximum distance to the closest points in the training set resulting in a discriminant function

$$f(\boldsymbol{x}) = \sum_{n=1}^{N} \alpha_n y_n \mathcal{K}(\boldsymbol{x}_n, \boldsymbol{x}) + \beta.$$

The classification result is then given by the sign of $f(\boldsymbol{x})$. The values of $\alpha_n$ and $\beta$ are found by solving a constrained minimization problem, which can be done efficiently using the SMO algorithm (Platt 1999). Most of the $\alpha_n$'s take the value of zero; those $\boldsymbol{x}_n$ with non-zero $\alpha_n$ are the "support vectors". In case where the two classes are non-separable, the optimization is formulated in such a way that the classification error is minimized and the final solution remains identical. The mapping between the input space and the usually high-dimensional feature space $\mathcal{H}$ is done using kernels $\mathcal{K}(\boldsymbol{x}_n, \boldsymbol{x})$.

The extension of SVM to multi-class problems can be done in several ways. Here we mention three approaches used throughout the paper:

1. *Standard one-against-all (OaA) strategy*. If $M$ is the number of classes, $M$ SVMs are trained, each separating a single class from all other classes. The decision is then based on the distance of the classified sample to each hyperplane, and the sample is assigned to the class corresponding to the hyperplane for which the distance is largest.

2. *Modified OoA strategy*. In Pronobis and Caputo (2007), a modified version of the OaA principle was proposed. The authors suggested to use distances to precomputed average distances of training samples to the hyperplanes (separately for each of the classes), instead of the distances to the hyperplanes directly. In this case, the sample is assigned to the class corresponding to the hyperplane for which the distance is smallest. Experiments presented in this paper and in Pronobis and Caputo (2007) show that in many applications this approach outperforms the standard OaA technique.

3. *One-against-one (OaO) strategy*. In this case, $M(M-1)/2$ two-class SVMs are trained for each pair of classes. The final decision can then be taken in different ways, based on the $M(M-1)/2$ outputs. A popular choice is to consider as output of each classifier the class label and count votes for each class; the test image is then assigned to the class that received more votes.

SVMs do not provide any out-of-the-box solution for estimating confidence of the decision; however, it is possible to derive confidence information and hypotheses ranking from the distances between the samples and the hyperplanes. In the work presented in this paper, we applied the distance-based methods proposed by Pronobis and Caputo (2007), which define confidence as a measure of unambiguity of the final decision related to the differences between the distances calculated for each of the binary classifiers.

### *4.2. Generalized Discriminative Accumulation Scheme*

G-DAS was first proposed by Pronobis and Caputo (2007), as a more effective generalization of the algorithm presented in Nilsback and Caputo (2004). It accumulates multiple cues, possibly from different modalities, by turning classifiers into experts. The basic idea is to consider real-valued outputs of a multi-class discriminative classifier (e.g. SVM) as an indication of a soft decision for each class. Then, all of the outputs obtained from the various cues are summed together, therefore linearly accumulated. Specifically, suppose we are given $M$ classes and, for each class, a set of $N_m$ training samples $\{\{\boldsymbol{s}_{m,n}\}_{n=1}^{N_m}\}_{m=1}^{M}$. Suppose also that, from each sample, we extract a set of $T$ different cues $\{\mathcal{T}_t(\boldsymbol{s}_{m,n})\}_{t=1}^{T}$. The goal is to perform recognition using all of them. The G-DAS algorithm consists of two steps:

1. *Single-cue Models.* From the original training set $\{\{\boldsymbol{s}_{m,n}\}_{n=1}^{N_m}\}_{m=1}^{M}$, containing samples belonging to all $M$ classes, define $T$ new training sets $\{\{\mathcal{T}_t(\boldsymbol{s}_{m,n})\}_{n=1}^{N_m}\}_{m=1}^{M}$, $t = 1, \ldots, T$, each relative to a single cue. For each new training set train a multi-class classifier. Then, given a test sample $\boldsymbol{s}$, for each of the $T$ single-cue classifiers estimate a set of outputs $\{\mathcal{V}_{t,v}(\mathcal{T}_t(\boldsymbol{s}))\}_{v=1}^{V}$ reflecting the relation of the sample to the model. In the case of the SVMs with standard OaO and OaA multi-class extensions, the outputs would be values of the discriminant functions learned by the SVM algorithm during training, i.e. $\mathcal{V}_{t,v}(\mathcal{T}_t(\boldsymbol{s})) = f_{t,v}(\mathcal{T}_t(\boldsymbol{s}))$, $v = 1, \ldots, V$, and $V = M(M-1)/2$ for OaO or $V = M$ for OaA.

2. *Discriminative Accumulation.* After all the outputs are computed for all the cues, combine them with different weights by a linear function:

$$\mathcal{V}_v(\boldsymbol{s}) = \sum_{t=1}^{T} \sigma_t \mathcal{V}_{t,v}(\mathcal{T}_t(\boldsymbol{s})), \ \sigma_t \in \Re^+, \ v = 1, \ldots, V.$$

   The final decision can be estimated with any method commonly used for multi-class, single-cue SVM.

An important property of accumulation is the ability to perform correct classification even when each of the single cues

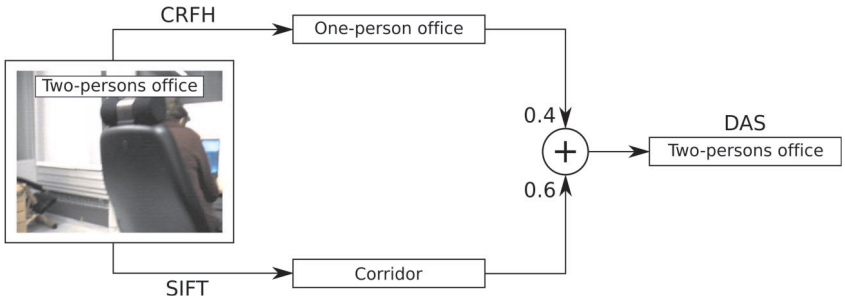



Fig. 2. A real example of test image misclassified by each of the single cues, but classified correctly using G-DAS.

gives misleading information. This behavior is illustrated on a real example in Figure 2. Despite these advantages, G-DAS presents some potential limitations: First, it uses only one weight for all outputs of each cue. This simplifies the parameter estimation process (usually, an extensive search is performed to find the coefficients $\{\sigma_t\}_{t=1}^{T}$), but also constrains the ability of the algorithm to adapt to the properties of each single cue. Second, accumulation is obtained via a linear function, which might not be sufficient in case of complex problems. The next section shows how our new accumulation scheme, SVM-DAS, addresses these issues.

### *4.3. SVM-based Discriminative Accumulation Scheme*

The SVM-DAS accumulates the outputs generated by single-cue classifiers by using a more complex, possibly non-linear function. The outputs are used as an input to an SVM, and the parameters of the integration function are learned during the optimization process, for instance using the SMO algorithm (Platt 1999). These characteristics address the potential drawbacks of G-DAS discussed in the previous section.

More specifically, the new SVM-DAS accumulation function is given by

$$\mathcal{V}_u(\boldsymbol{s}) = \sum_{n=1}^{N} \alpha_{u,n} y_n \mathcal{K}(\boldsymbol{v}_n, \boldsymbol{v}) + \beta_u, \quad u = 1, \ldots, U,$$

where $\boldsymbol{v}$ is a vector containing all the outputs for all $T$ cues:

$$\boldsymbol{v} = \left[ \{\mathcal{V}_{1,v}(\mathcal{T}_1(\boldsymbol{s}))\}_{v=1}^{V_1}, \ldots, \{\mathcal{V}_{T,v}(\mathcal{T}_T(\boldsymbol{s}))\}_{v=1}^{V_T} \right].$$

The parameters $\alpha_{u,n}$, $y_n$, $\beta_u$, and the support vectors $\boldsymbol{v}_n$ are inferred from the training data either directly or efficiently during the optimization process. The number of the final outputs $U$ and the way of obtaining the final decision depends on the multi-class extension used with SVM-DAS. We use the OaO extension throughout the paper for which $U = M(M-1)/2$.

The non-linearity is given by the choice of the kernel function $\mathcal{K}$, thus in the case of the linear kernel the method is still linear. In this sense, SVM-DAS is more general than G-DAS, while it preserves all of its important properties (e.g. the ability to give correct results for two misleading cues, see Figure 2). Also, for SVM-DAS each of the integrated outputs depend on all the outputs from single-cue classifiers, and the coefficients are learned optimally. Note that the outputs $\mathcal{V}_{t,v}(\mathcal{T}_t(\boldsymbol{s}))$ can be derived from a combination of different large margin classifiers, and not only from SVM[4].

## 5. Place Classification for Semantic Space Labeling

One of the applications of a place classification system is semantic labeling of space. This section provides a brief overview of the problem and describes how we employed our multi-modal place classification method to build a semantic labeling system. We evaluated the system in a live experiment described in Section 7.

### *5.1. Semantic Labeling of Space*

The problem of semantic labeling can be described as assigning meaningful semantic descriptions (e.g. "corridor" or "kitchen") to areas in the environment. Typically, semantic labeling is used as a way of augmenting the internal space representation with additional information. This can be used by the agent to reason about space and to enhance communication with a human user. In case of most typical environments, it is sufficient to distinguish between semantic categories which are usually associated with rooms (Zender et al. 2007), such as "office", "meeting room" or "corridor". It is labeling at this level that we will discuss in this paper.

4. SVM-DAS can be seen as a variation of ensemble learning methods that employ multiple models to improve the recognition performance. The key reason why ensemble algorithms obtain better results is because the individual classifiers make errors on different data points. Typically, different training data is used for each classifier (Polikar 2006). In our experiments, we use data representing different types of information, e.g. obtained using different sensors.

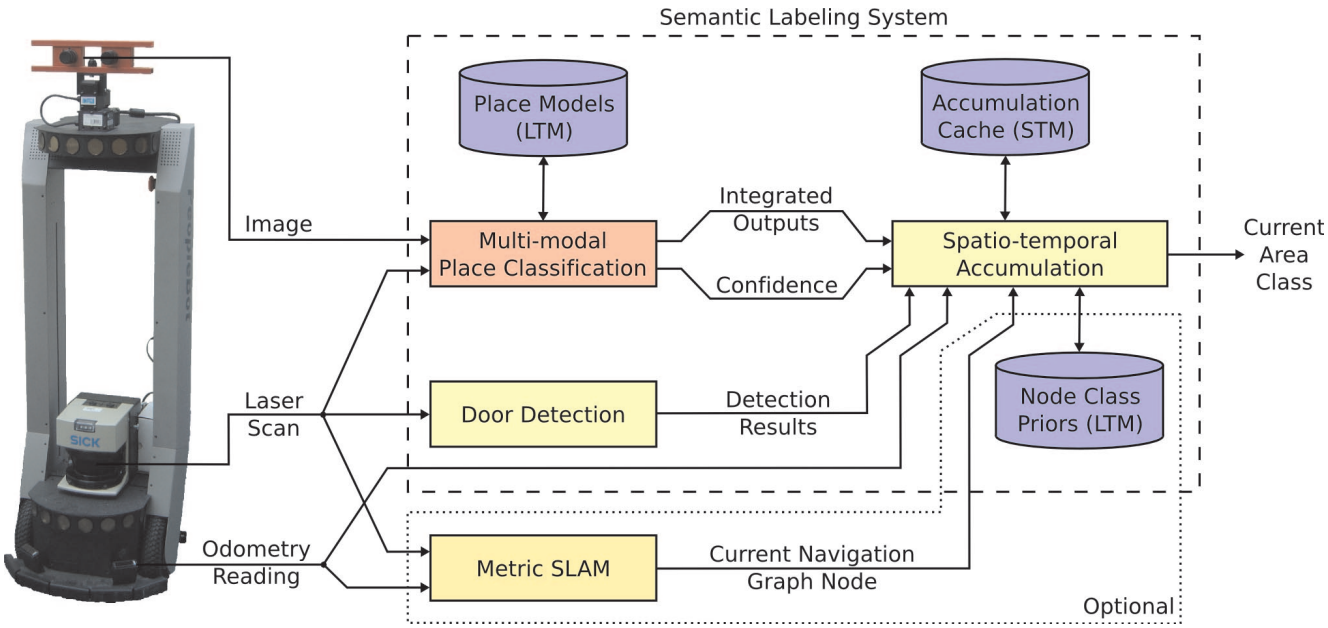



Fig. 3. Architecture of the semantic space labeling system based on place classification (LTM: Long-Term Memory; STM: Short-Term Memory).

As will be shown through experiments in Sections 6 and 7, the place classification system described in this paper can yield a place class with high accuracy given a single sample of multi-modal data (e.g. one image and a laser scan). However, when used for semantic labeling, the algorithm is requested to provide a label for the whole area under exploration. At the same time, the system must be resilient and able to deal with such problems as temporary lack of informative cues, continuous stream of similar information or long-term occlusions. Given that the system is operating on a mobile robot, crude information about its movement is available from wheel encoders. This information can be used to ensure robustness to the typical variations that occur in the environment but also to the problems mentioned above. Finally, the system should be able to measure its own confidence and restrain from making a decision until some confidence level is reached. All of these assumptions and requirements have been taken into consideration while designing the system described in the following section.

### *5.2. Architecture of the System*

The architecture of our system is presented in Figure 3. The system relies on three sensor modalities typically found on a mobile robot platform: a monocular camera, a single 2D laser scanner, and wheel encoders. The images from the camera, together with the laser scans are used as an input for the multi-modal place classification component. For each pair of data samples, place classification provides its beliefs about the semantic category to which the samples belong. These beliefs are encoded in the integrated outputs as discussed in Section 4. Moreover, the confidence of the decision is also measured and provided by the classification component.

A labeling system should provide a robust and stable output over the whole area. Since the sensors employed are not omni-directional, it is necessary to accumulate and fuse information over time. Moreover, the data that the robot gathers are not evenly spread over different viewpoints. As a possible solution, we propose to use a confidence-based spatio-temporal accumulation method. The principle behind the method is illustrated in Figure 4. As the robot explores the environment, it moves with a varying speed. The robot has information about its own movement (odometry) provided by the wheel encoders. As errors accumulate over time, this information can only be used to estimate relative movement rather than absolute position. This is sufficient for our application. The spatio-temporal accumulation process creates a sparse histogram along the robot pose trajectory given by the odometry and described by the metric position $(x, y)$ and heading $(\theta)$ as shown in Figure 4. The size of the histogram bins are adjusted so that each bin roughly corresponds to a single viewpoint. Then, as the robot moves, the beliefs about the current semantic category accumulate within the bins as in the case of G-DAS (with equal weights). This is what we call the temporal accumulation. It prevents a single viewpoint from becoming dominant due to long-term observation. Since each viewpoint observed by the robot will correspond to a different bin, performing accumulation across the bins (this time spatially) allows to generate the final outputs to which each viewpoint contributes equally. In order to exclude most of the misclassifications before they get

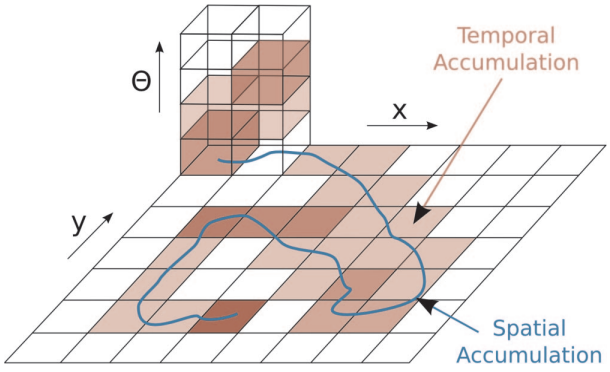



Fig. 4. Illustration of the spatio-temporal accumulation process. As the robot explores the environment, the beliefs collected on the way accumulate over time within the bin corresponding to the current pose $(x, y, \theta)$ and over space in different bins.

accumulated, we filter the decisions based on the confidence value provided by the place classification component. Moreover, as the odometry information is unreliable in the long term, the contents of bins visited a certain amount of viewpoints ago, are invalidated. Note that semantic labeling is an application of the method presented in this paper and not the main focus of the paper. The accumulation scheme we present here builds on the ideas of discriminative accumulation and confidence estimation to further illustrate their usefulness. If the emphasis was on labeling, more advanced methods based on Hidden Markov Models (Rottmann et al. 2005), probabilistic relaxation (Stachniss et al. 2007) or Conditional Random Fields (Douillard et al. 2007) should be taken into consideration. The advantages of our method are seamless integration with other components of the system and simplicity (the method does not require training or making assumptions on the transition probabilities between locations or areas).

The accumulation process ensures robustness and stability of the generated label for a single area. However, another mechanism is required to provide the system with information about area boundaries. This is required for the accumulation process not to fuse the beliefs across different areas. Here, we propose two solutions to that problem. As described in the previous sections, we can assume that each room of the environment should be assigned one semantic label. In the case of indoor environments, rooms are usually separated by a door or other narrow openings. Thus, as one solution, we propose to use a simple laser-based door detector which generates hypotheses about doors based on the width of the opening which the robot passes. Such a simple algorithm will surely generate a lot of false positives. However, this does not cause problems in the presented architecture as false positives only lead to oversegmentation. This is a problem mainly for other components relying on precise segmentation rather than for the labeling process itself. In fact, the labeling system could be used to identify false doors and improve the segmentation by looking for directly connected areas classified as being of the same category.

As a second solution, we propose to use another localization and mapping system in order to generate the space representation which will then be augmented with semantic labels. Here we take the multi-layered approach proposed in Zender et al. (2008). The method presented by Zender et al. (2008) builds a global metric map as the first layer and a navigation graph as the second. As the robot navigates through the environment, a marker or navigation node is dropped whenever the robot has traveled a certain distance from the closest existing node. Nodes are connected following the order in which they were generated. If information about the current node is provided to the spatio-temporal accumulation process, labels can be generated for each of the nodes separately. Moreover, as it is possible to detect whether the robot revisited an existing node, the accumulated information can be saved and used as a prior the next time the node is visited. For the live experiment described in this paper, we used the detected doors to bound the areas and navigation graph nodes to keep the priors. We then propagated the current area label to all the nodes in the area.

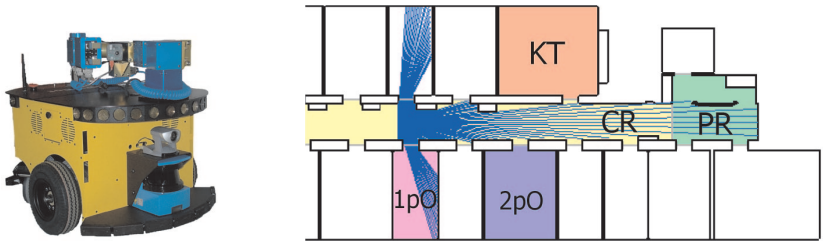
## 6. Experiments with Place Classification

We conducted several series of experiments to evaluate the performance of our place classification system. We tested its robustness to different types of variations, such as those introduced by changing illumination or human activity over a long period of time. The evaluation was performed on data acquired using a mobile robot platform over a time span of six months, taken from the IDOL2 database (Image Database for rObot Localization 2, see Luo et al. (2007)). Details about the database and experimental setup are given in Section 6.1. The experiments were performed for single-cue models and models based on different combinations of cues and modalities. We present the results in Sections 6.2 and 6.3 respectively. In addition, we analyze performance and properties of different cue integration schemes in Section 6.4.

### *6.1. Experimental Setup*

The IDOL2 database was introduced in Luo et al. (2007). It comprises of 24 image sequences accompanied by laser scans and odometry data acquired using two mobile robot platforms (PeopleBot and PowerBot). The images were captured with a Canon VC-C4 perspective camera using the resolution of 320 × 240 pixels. In this paper, we will use only the 12 data sequences acquired with the PowerBot, shown in Figure 5(a).

The acquisition was performed in a five room subsection of a larger office environment, selected in such a way that each of the five rooms represented a different functional area: a




(a) (b)

Fig. 5. (a) The mobile robot platform used in the experiments. (b) Map of the environment used during data acquisition and an example laser scan simulated in the corridor. The rooms used during the experiments are annotated.

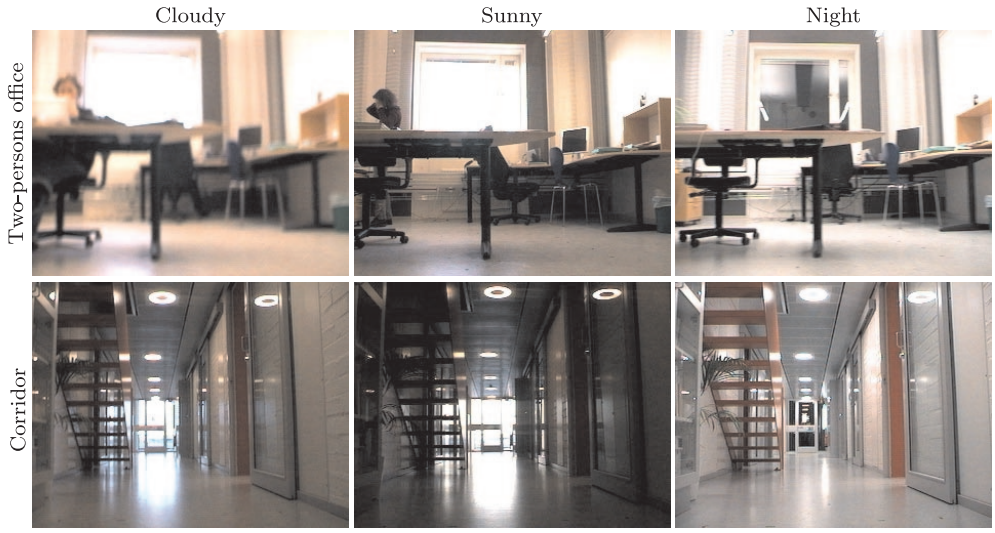
one-person office (1pO), a two-persons office (2pO), a kitchen (KT), a corridor (CR), and a printer area (PR). The map of the environment and an example laser scan are shown in Figure 5(b). Example pictures showing interiors of the rooms are presented in Figure 6. The appearance of the rooms was captured under three different illumination conditions: in cloudy weather, in sunny weather, and at night. The robot was manually driven through each of the five rooms while continuously acquiring images and laser range scans at a rate of 5 fps. Each data sample was then labelled as belonging to one of the rooms according to the position of the robot during acquisition. Extension 1 contains a video illustrating the acquisition process of a typical data sequence in the database. The acquisition was conducted in two phases. Two sequences were acquired for each type of illumination conditions over the time span of more than two weeks, and another two sequences for each setting were recorded six months later (12 sequences in total). Thus, the sequences captured variability introduced not only by illumination but also natural activities in the environment (presence/absence of people, furniture/objects relocated etc.). Example images illustrating the captured variability are shown in Figure 6.

We conducted four sets of experiments, first for each cue separately and then for cues combined. In order to simplify the experiments with multiple cues, we matched images with closest laser scans on the basis of the acquisition timestamp. In case of each single experiment, both training and testing were performed on one data sequence. The first set consisted of 12 experiments, performed on different combinations of training and test data acquired closely in time and under similar illumination conditions. In this case, the variability comes from human activity and viewpoint differences. For the second set of experiments, we used 24 pairs of sequences captured still at relatively close times, but under different illumination conditions. In this way, we increased the complexity of the problem (Pronobis et al. 2006; Pronobis and Caputo 2007). In the third set of experiments, we tested the robustness of the system to long-term variations in the environment. Therefore, we conducted 12 experiments, where we tested on data acquired six months later, or earlier, than the training data, again under similar illumination conditions. Finally, we combined both types of variations and performed experiments on 24 pairs of training and test sets, obtained six months from each other and under different illumination settings. Note that in the last two sets of experiments described, the task becomes more and more challenging as the difference between training and test set increases. By doing this, we aim at testing the gain in robustness expected from cue integration in very difficult, but still realistic, scenarios.

For all experiments, model parameters were determined via cross validation. Since the datasets in the IDOL2 database are unbalanced (on average 443 samples for CR, 114 for 1pO, 129 for 2pO, 133 for KT and 135 for PR), as a measure of performance for the reported results and parameter selection, we used the average of classification rates obtained separately for each actual class (average per-class recall). For each single experiment, the percentage of properly classified samples was first calculated separately for each room and then averaged with equal weights independently of the number of samples acquired in the room. This allowed to eliminate the influence that large classes could have on the performance score. Statistical significance of the presented results was verified using the Wilcoxon signed-ranks test (when performance of two methods was compared) or Friedman and *post hoc* Nemenyi test (when multiple methods were compared) at a confidence level of $\alpha = 0.05$ as suggested in Demšar (2006). The results of the *post hoc* tests were visualized using critical difference diagrams. The diagrams show average ranks of the compared methods and the groups of methods that are not significantly different are connected (the difference is smaller than the critical difference presented above the main axis of the diagram). The reader is referred to Demšar (2006) for details on the applied tests and the critical difference diagrams presented below.

### *6.2. Experiments with Separate Cues*

We first evaluated the performance of all four types of single-cue models: the two SVM models based on visual features




(a) Variations introduced by illumination


(b) Variations observed over time


(c) Remaining rooms (at night)

Fig. 6. Examples of pictures taken from the IDOL2 database showing the interiors of the rooms, variations observed over time and caused by activity in the environment as well as introduced by changing illumination.

(CRFH, SIFT), the AdaBoost and the SVM models trained on the laser range cues (referred to as L-AB and L-SVM). For SVM, we tried the three multi-class extensions described in Section 4.1. The results of all four sets of experiments for these models are presented in Figures 7–10 (the first four bar groups). Moreover, the results of statistical significance tests comparing the models based on the combined results of all four experiments are illustrated in Figure 11. We first note that,

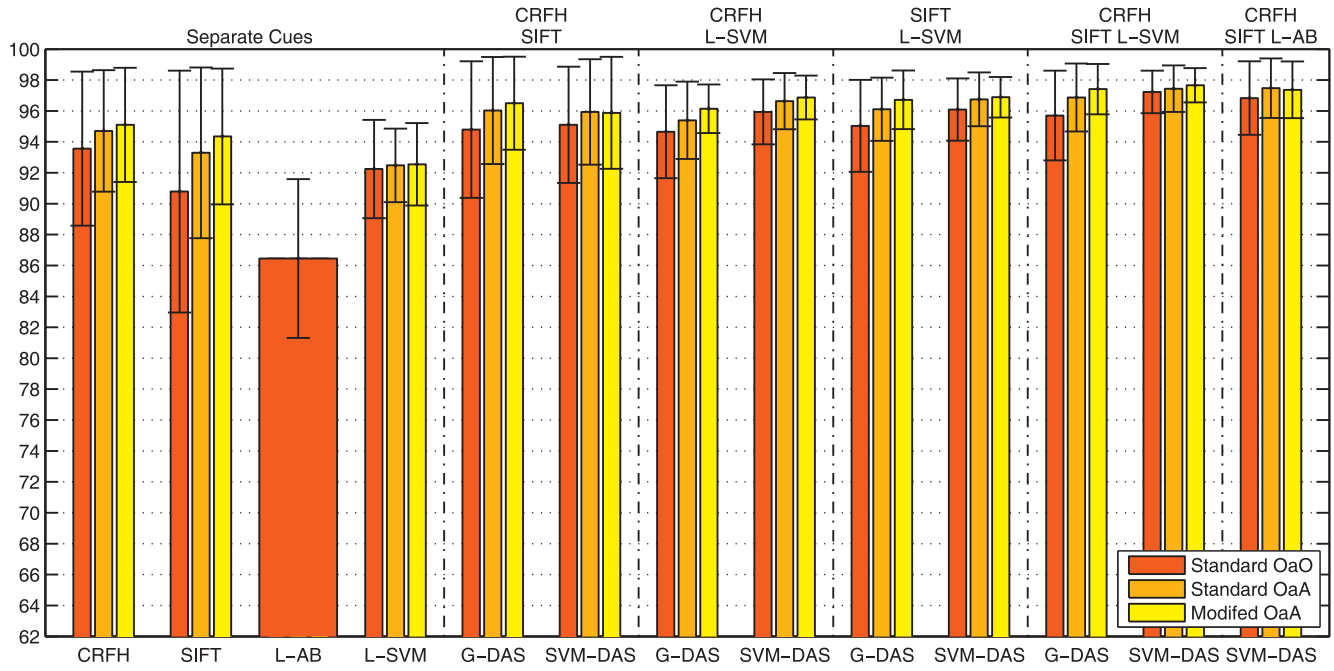



Fig. 7. Classification results for Experiment 1: stable illumination conditions, close in time.

as expected, CRFH and SIFT suffer from changes in illumination (−15.3% and −11.0% respectively), while the geometrical features do not (−1.9% for L-AB and −0.64% for L-SVM). Long-term variations pose a challenge for both modalities (from −7.0 to −10.2% for vision and −3.7 to −7.9% for laser). We also see differences in performance between the two visual cues: CRFH suffers more from changes in illumination, while SIFT is less robust to variations induced by human activities. It is also interesting to note that under stable conditions, the vision-based methods outperform the systems based on laser range cues (95.1% for CRFH and 92.5% for L-SVM; the difference is statistically significant). This illustrates the potential of visual cues, but also stresses the need for more robust solutions.

These experiments are also a comparison between two recognition algorithms using laser-range features, namely the boosting-based implementation (L-AB) presented in Mozos et al. (2005) and the current SVM-based implementation (L-SVM). Figures 7–10 and Figure 11 show the results. We can see that the difference in performance is statistically significant in favor of the SVM-based method for all three multi-class extensions (from 6.1% for Experiment 1 to 10.3% for Experiment 4 in average). The classification results for the L-AB are worse than the results of the original paper by Mozos et al. (2005). There are two main reasons for that. First, the number of classes is increased to five, while in Mozos et al. (2005) was of a maximum of four. Second, in these experiments, we used a restricted field of view of 180°, whilst in Mozos et al. (2005) the field of view was of 360°. This decreases the classification rate, as has been shown in previous work (Mozos et al. 2007).

As already mentioned, all the experiments with SVMs were repeated for three different multi-class extensions: standard OaO and OaA as well as modified OaA algorithm. The obtained results are in agreement with those of Pronobis and Caputo (2007): in the case of single cue and G-DAS experiments, the modified version gives the best performance with a statistically significant difference independently of the modality on which the classifier was trained.

Figure 12 shows the distribution of errors for each actual class (room) made by the four models. It is apparent that each of the cues makes errors according to a different pattern. At the same time, similarities occur between the same modalities. We see that visual models are biased towards the corridor, while the geometrical models tend to misclassify places as the printer area. A possible explanation is that the vision-based models were trained on images acquired with perspective camera with constrained viewing angle. As a result, similar visual stimuli coming from the corridor are present in the images captured by the robot leaving each of the rooms. The same area close to a doorway, from the geometrical point of view, is similar to the narrow passage in the printer area. This analysis is a strong motivation to integrate these various cues with a stack of classifiers, as theory indicates that this is the ideal condition for exploiting the different informative content (Polikar 2006).

### *6.3. Experiments with Cue Integration*

For the final experiments, we selected four different cue accumulation methods: G-DAS and SVM-DAS with three kernel types (linear, RBF, and histogram intersection (HI) kernel

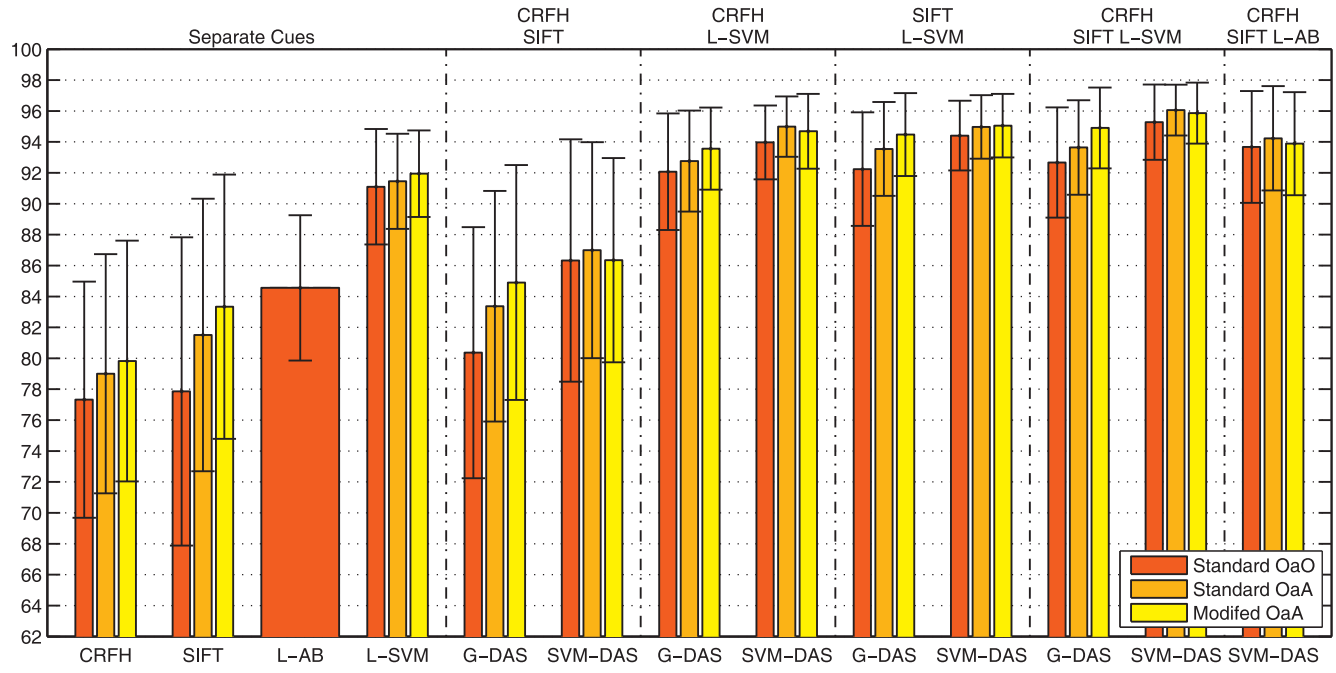



Fig. 8. Classification results for Experiment 2: varying illumination conditions, close in time.

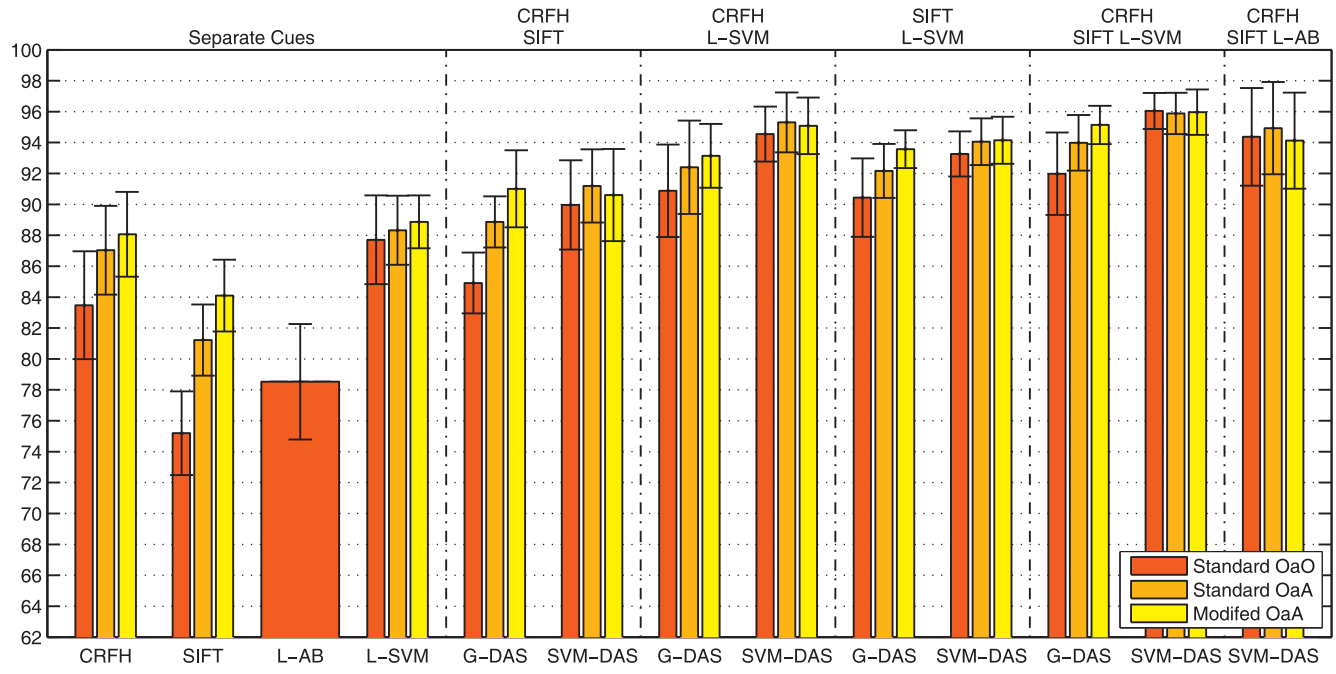



Fig. 9. Classification results for Experiment 3: stable illumination conditions, distant in time.

(Barla et al. 2003)). The parameters of the algorithms (weights in case of G-DAS and SVM model in case of SVM-DAS) were always adjusted on the basis of outputs generated during all experiments with single-cue models trained on one particular data sequence. Then, during testing, the previously obtained integration scheme was applied to all experiments with models trained on a different sequence, acquired under similar illumination and closely in time. This way, the generalization abilities of each of the methods were tested in a realistic scenario. In all experiments, we found that SVM-DAS with an RBF ker-

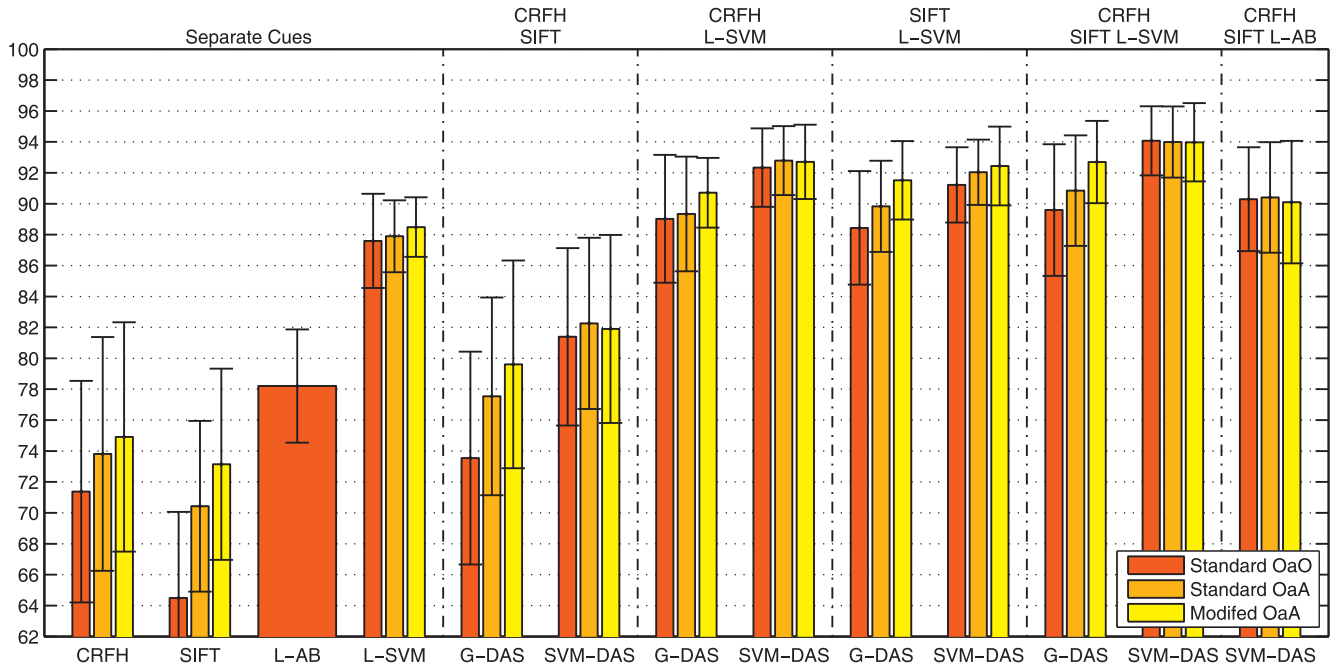



Fig. 10. Classification results for Experiment 4: varying illumination conditions, distant in time.

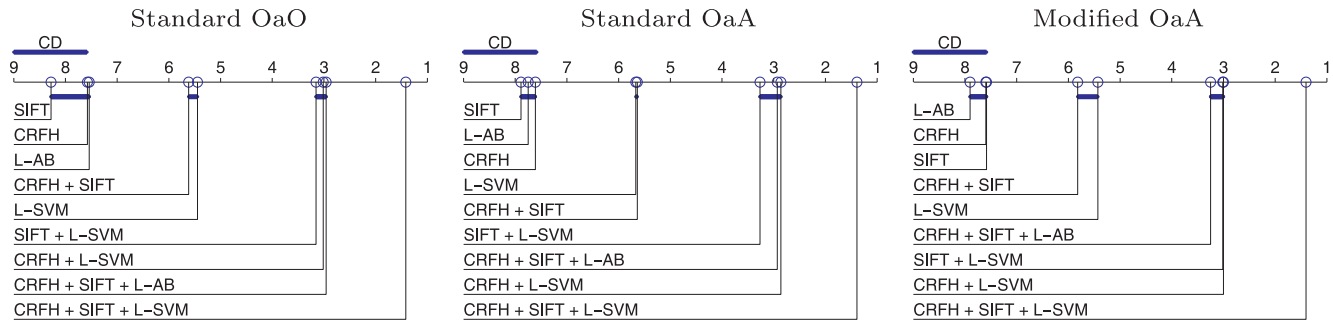



Fig. 11. Critical difference diagrams comparing four single-cue models and solutions based on multiple cues integrated using SVM-DAS with the Nemenyi test for a confidence level of $\alpha = 0.05$. The comparison is based on the combined results of Experiments 1–4 and presented separately for each multi-class extension. The average ranks of the methods are plotted on the axis and the groups of methods that are not significantly different are connected.

nel outperforms the other methods and the difference in performance with respect to G-DAS was statistically significant for all combinations of cues and multi-class extensions (Wilcoxon test). For space reasons, we report results of each of the experiments only using SVM-DAS based on the RBF kernel and G-DAS for comparison (Figures 7–10, last nine bar groups). A detailed comparison of all variants of SVM-DAS for the most complex problem (Experiment 4) is given in Figure 13. Results of statistical significance tests comparing the multi-cue solutions with single-cue models based on the combined results of all experiments are illustrated in Figure 11.

We tested the methods with several combinations of different cues and modalities. First, we combined the two visual cues. We see that the generalization of a purely visual recognition system can be significantly improved by integrating different types of cues, in this case local and global. This can be observed especially for Experiment 4, where the algorithms had to tackle the largest variability. Despite that, according to the error distributions in Figure 12, we should expect the largest gain when different modalities are combined. As we can see from Figures 7–10 this is indeed the case. By combining one visual cue and one laser range cue (e.g. CRFH + L-SVM), we exploit the descriptive power of vision in the case of stable illumination conditions and the invariance of geometrical features to the visual noise. Moreover, if the computational cost is not an issue, the performance can be further improved by

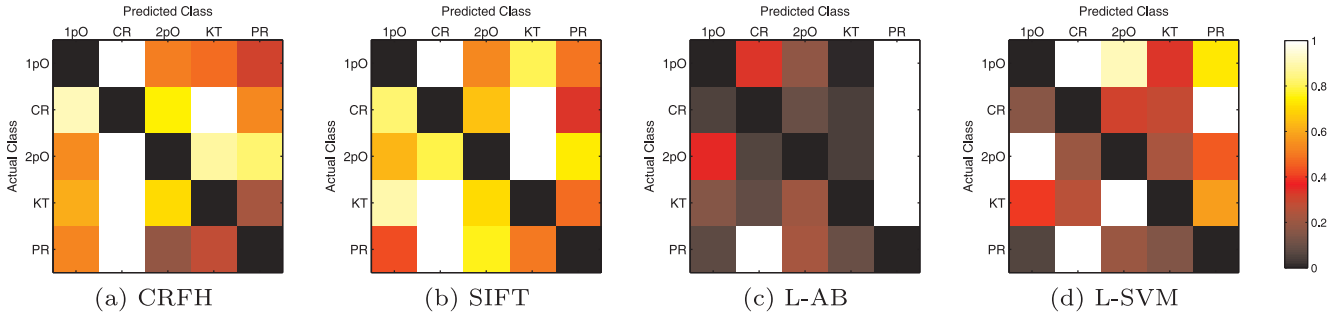



Fig. 12. Distribution of errors made by the four models for each actual class (bright colors indicate errors). The diagonal elements were removed.

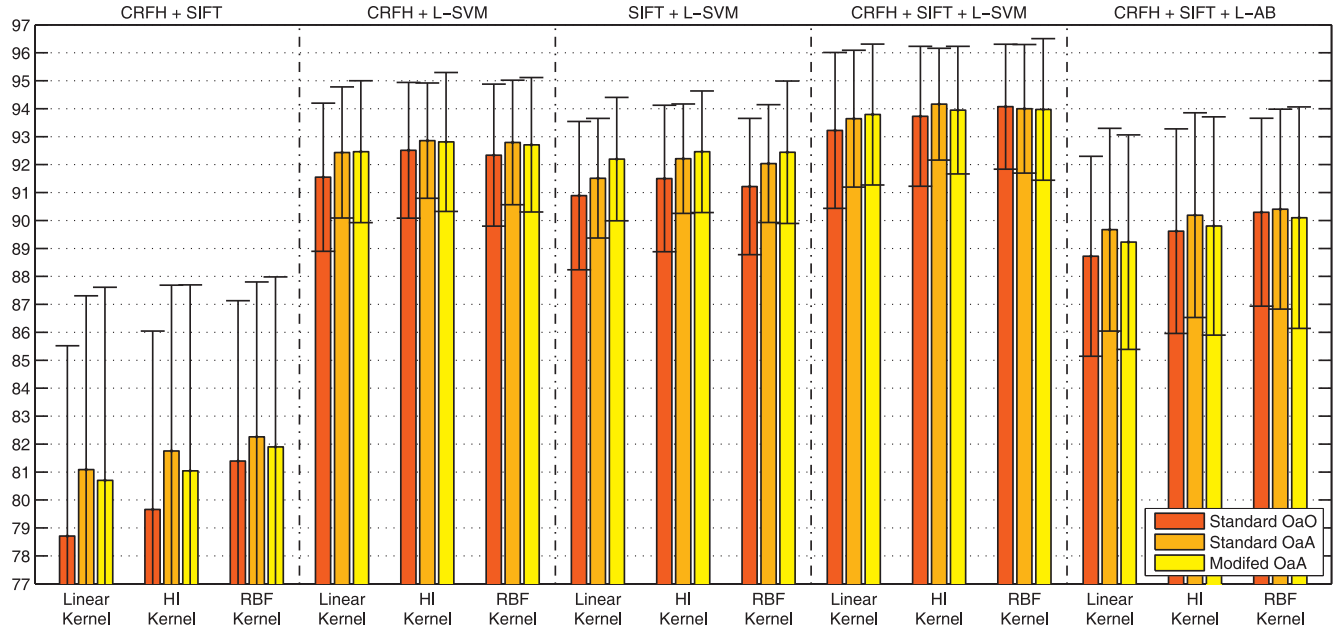



Fig. 13. Comparison of performance of SVM-DAS based on different kernel functions for the most complex problem (Experiment 4).

using both visual cues instead of just one. As can be seen from Figure 11, by integrating single-cue models or adding another cue to a multi-cue system, we always get an improvement statistically significant.

We performed a more detailed analysis of the best results. Table 1 contains the confusion matrix for the multi-cue system based on CRFH, SIFT and L-SVM integrated using SVM-DAS with an RBF kernel. We see that even if the corridor class contained on average four times more samples than each of the room classes and was visually and geometrically distinctive, the results are balanced and the recognition rates for each actual class are similar. In general, during our experiments, more balanced solutions were preferred due to the performance metric used (average of the diagonal values in brackets in Table 1).

As it was mentioned in Section 4.3, SVM-DAS can be applied for problems where outputs of different classifiers need to be integrated. To test this in practice, we combined the SVM models trained on visual cues with AdaBoost model based on geometrical features (L-AB)[5]. We present the results in Figures 7–10 (last bar group) and Figure 11. The method obtained large and statistically significant improvements compared to each of the individual cues. For instance for Experiment 4, the recognition rate in-

5. As usual, for SVM we used several multi-class extensions that in most cases produced outputs having different interpretation than those generated by the multi-class algorithm used for AdaBoost. In those cases G-DAS could not be applied.

**Table 1. Confusion Matrix for the Multi-cue System Based on CRFH, SIFT and L-SVM Integrated Using SVM-DAS**

| | Predicted class | | | | |
|---|---|---|---|---|---|
| Actual class | 1pO | CR | 2pO | KT | PR |
| 1pO | **11.20 (93.71)** | 0.36 (3.06) | 0.16 (1.33) | 0.11 (0.96) | 0.11 (0.94) |
| CR | 0.25 (0.53) | **45.36 (97.73)** | 0.19 (0.42) | 0.33 (0.70) | 0.29 (0.62) |
| 2pO | 0.17 (1.22) | 0.11 (0.8) | **13.26 (96.92)** | 0.06 (0.46) | 0.08 (0.60) |
| KT | 0.17 (1.18) | 0.35 (2.45) | 0.08 (0.57) | **13.42 (95.12)** | 0.09 (0.67) |
| PR | 0.09 (0.65) | 0.77 (5.59) | 0.03 (0.19) | 0.05 (0.33) | **12.90 (93.24)** |

Normalized average values in percentage over all experiments are reported. The values in brackets were normalized separately for each actual class (row). The presented results are only for the standard OaO multi-class extension since the results for the remaining extensions were comparable.

creased by 12.2% in average. This proves the versatility of our approach.

### *6.4. Analysis of Cue Integration Schemes*

Results presented so far clearly show that SVM-DAS performs significantly better than G-DAS and, by using more sophisticated kernel types for SVM-DAS, it is possible to perform non-linear cue accumulation. Moreover, the experiments (see Figure 13) show that we can expect better results with the RBF kernel (especially for the OaO multi-class extension), although there is no drastic improvement. We therefore suggest to choose the kernel according to constraints on the computational cost of the solution. Since there are fast implementations of linear SVMs, it might be beneficial to use a linear kernel in cases when the integration scheme must be trained on a very large number of samples. In applications where only the number of training parameters is an issue, the non-parametric HI kernel can be used instead of RBF.

We now further discuss differences between high-level (e.g. SVM-DAS) and low-level (feature-level) cue integration. There are several advantages in integrating multiple cues with a high-level strategy. First, different learning algorithms can be used for each single cue. In our experiments, this allowed to combine SVM-based models employing different kernel functions (e.g. the $\chi^2$ kernel for CRFH and the match kernel for SIFT) or even different classifiers (AdaBoost and SVM). Moreover, parameters can be tuned separately for each of the cues. Second, both the training and recognition tasks can be divided into smaller subproblems that can be easily parallelized. Finally, it is possible to decide on the number of cues that should be extracted and used for each particular classification task. This is an important feature, since, in most cases, decisions based on a subset of cues are correct while extraction and classification of additional features introduces additional cost. For example, a solution based on global visual features, laser range cues and SVM-DAS runs in real-time at a rate of approximately 5 fps, which would not be possible if an additional visual cue like SIFT was used. The computational cost

**Table 2. Average Percentages (with Standard Deviations) of Test Samples for which all Cues had to be Used in Order to Obtain the Maximal Recognition Rate**

| | Cue integration method | |
|---|---|---|
| Cues (**Primary cue**) | G-DAS | SVM-DAS RBF Kernel |
| **CRFH** + SIFT | $25.971 \pm 18.503$ | $29.453 \pm 22.139$ |
| CRFH + **L-SVM** | $21.230 \pm 20.199$ | $32.736 \pm 20.256$ |
| SIFT + **L-SVM** | $28.820 \pm 20.982$ | $33.344 \pm 22.425$ |
| SIFT + CRFH + **L-SVM** | $31.858 \pm 20.474$ | $40.833 \pm 21.916$ |

can be significantly reduced by taking the approach presented in Pronobis and Caputo (2007). By combining confidence estimation methods with cue integration, we can use additional sources of information only when necessary – when the decision based on one cue only is not confident enough. This scheme is referred to as Confidence-based Cue Integration. Table 2 presents the results of applying the scheme to the experiments presented in this section. We see that, in general, we can base our decision on the fastest model (marked with bold font in Table 2), such as the efficient and low-dimensional model based on simple laser-range features, and we can retain the maximal performance by using additional cues only in approximately 30% of cases. This greatly reduces the computational time required on average e.g. approximately three times for CRFH, L-SVM and SVM-DAS. Additional cues will be used more often when the variability is large, and rarely for less difficult cases. This is not possible in the case of low-level integration where all the cues must be extracted and classified in order to obtain a decision.

Another important factor is performance. During our experiments, we compared the performance of G-DAS and SVM-DAS (with an RBF kernel) with models built on cues combined on the feature level. We performed three different sets

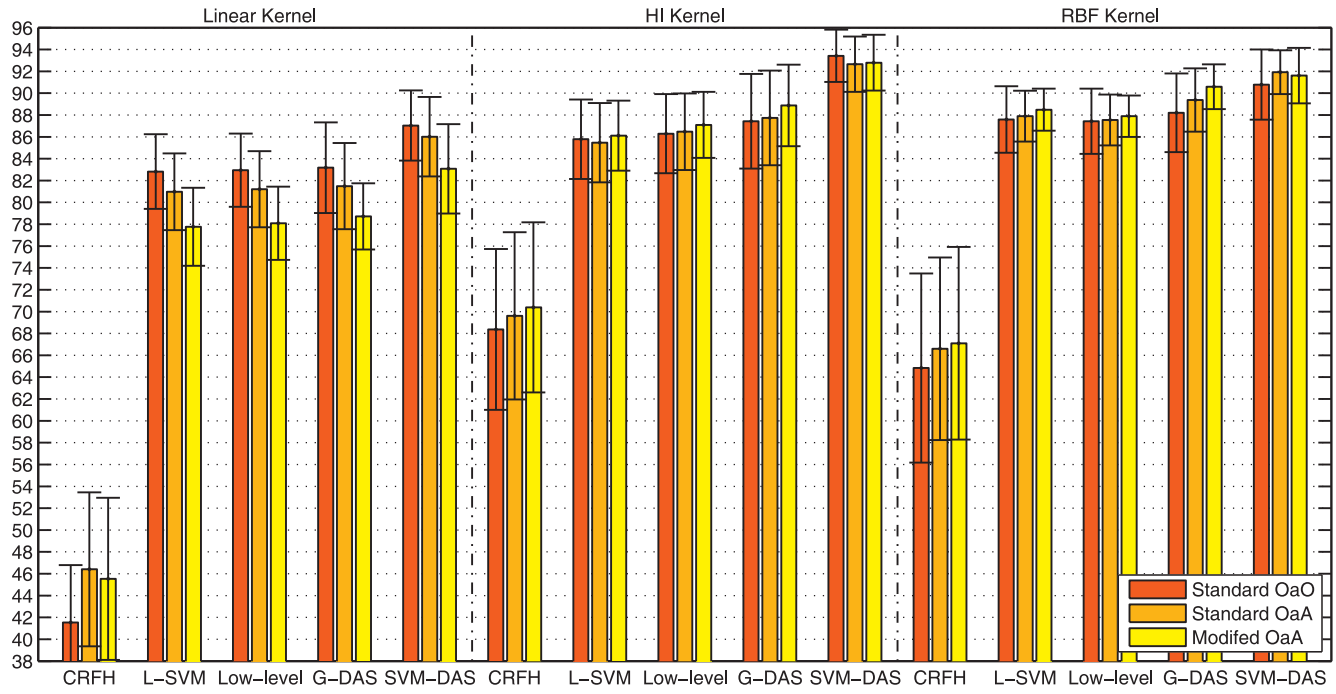



Fig. 14. Comparison of performance of two single-cue models and solutions based on the cues integrated on both low and high level for the most complex problem (Experiment 4).

of comparisons. In the first comparison, we built single-cue models and models based on features combined on the low level using SVM and the non-parametric linear kernel, using the same values of the SVM training parameters for all models. Then, we integrated the outputs of the single-cue models using G-DAS and SVM-DAS. In the case when G-DAS was used, the solution remained linear. In the second comparison, for building the models we used the non-linear, non-parametric HI kernel. In the final comparison, we used an RBF kernel and performed parameter selection for each of the models. All comparisons were based on CRFH and laser-range cues, since the dedicated kernel function required by SIFT could not be used with any of the other features for low-level integration.

The results for the most complex problem (Experiment 4) are given in Figure 14 and statistical significance tests comparing the solutions are illustrated in Figure 15. It can be observed that, in every case, the high-level integration significantly outperformed solutions based on features combined on the low level. In only one case there was no significant difference between G-DAS and low-level integration; however, SVM-DAS still performed better than the other solutions. This is in agreement with the results reported by Tommasi et al. (2008) and Nilsback and Caputo (2004) and can be explained by greater robustness of the high-level methods to noisy cues or sensory channels and the ability of different classifiers to adapt to the characteristics of each single cue.

## 7. Experiments with Semantic Space Labeling

We performed an independent live experiment to test our multi-modal semantic space labeling system running in real-time on a mobile robot platform. The experiment was performed during working hours in a typical office environment. Both the environment and the robot platform were different than in the case of the off-line evaluation described in Section 6. The whole experiment was videotaped and a video presenting the setup, experimental procedure, and visualization of the results can be found in Extension 2.

### *7.1. Experimental Setup*

The experiment was performed between the 7th and 10th of September 2008 in the building of the School of Computer Science at the University of Birmingham, Birmingham, UK. The interior of the building consists of several office environments located on three floors. For our experiments, we selected three semantic categories of rooms that could be found in the building: a corridor, an office and a meeting room. To build the model of an office, we acquired data in three different offices: Aaron's office (first floor), Robert's office (first floor) and Richard's office (ground floor). To create a representation of the corridor class, we recorded data in two corridors, one on the ground floor and one on the first floor. The acquisition

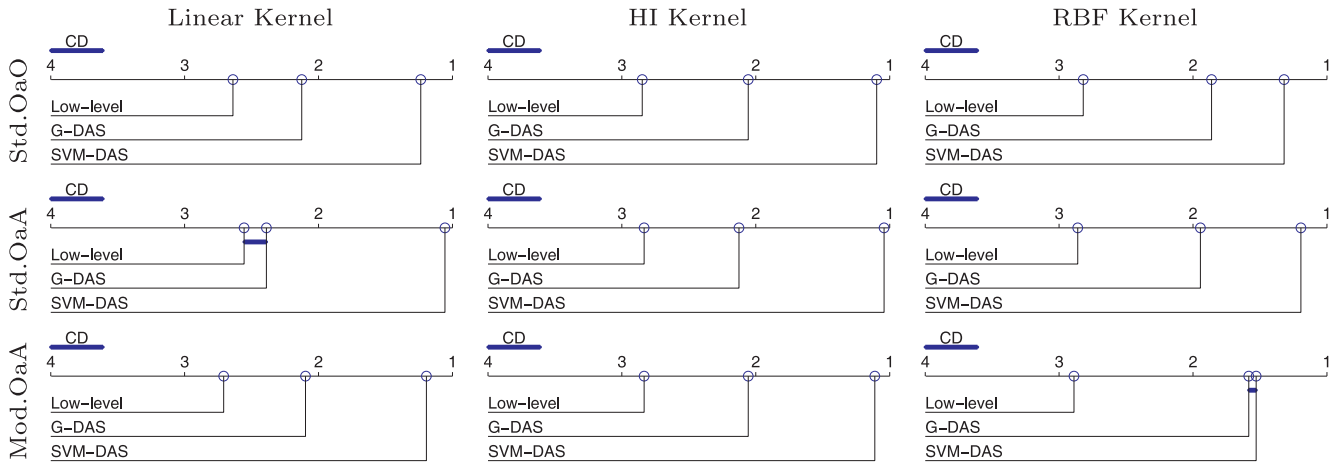



Fig. 15. Critical difference diagrams comparing two single-cue models and solutions based on the cues integrated at both the low and high level with the Nemenyi test for a confidence level of $\alpha = 0.05$. The comparison is based on the combined results of Experiments 1–4 and presented separately for three kernel functions and multi-class extensions used with SVM. The average ranks of the methods are plotted on the axis and the groups of methods that are not significantly different are connected.

was performed at night. Finally, to train the model of a meeting room, we used an instance on the second floor. All training data except the one from the meeting room was acquired in another part of the building than the one used for testing. The data for this class were recorded during the day. A video illustrating the whole data acquisition process is available as Extension 3. The interiors of the rooms are presented in Figure 16(a), as seen by vision and laser. The robot was manually driven around each room and data samples were recorded at the rate of 5 fps. All the collected training data are available as Extension 4. In the case of the meeting room, the corridor on the first floor as well as Aaron's and Richard's offices, the acquisition was repeated twice.

For the real-time experiment, we built the system as described in Section 5. Following the findings of the off-line experiments, we used SVM-DAS with the RBF kernel to integrate the classifier outputs for vision and laser range data. For efficiency reasons, we used only global features (CRFH) for the vision channel. We used the OaA multi-class SVM extension for the place models. Other parameters were set as described in Section 6.

We trained the place models separately for each modality on a dataset created from one data sequence recorded in each of the rooms. One of the advantages of SVM-DAS is the ability to infer the integration function from the training data, after training the models. We used the additional data sequences acquired in some of the rooms and trained SVM-DAS on the outputs of the uni-modal models tested on these data.

The PeopleBot robot platform shown in Figure 3 was used for data acquisition and the final experiment. The robot was equipped with a SICK laser range finder and Videre STH-MDCS2 stereo head (only one of the cameras was used). The images were acquired at the resolution of $320 \times 240$ pixels. The whole system was implemented in the CAST (The CoSy Architecture Schema Toolkit)[6] framework and run on a standard 2.5 GHz dual-core laptop. The processing for both modalities was executed in parallel using both of the CPU cores.

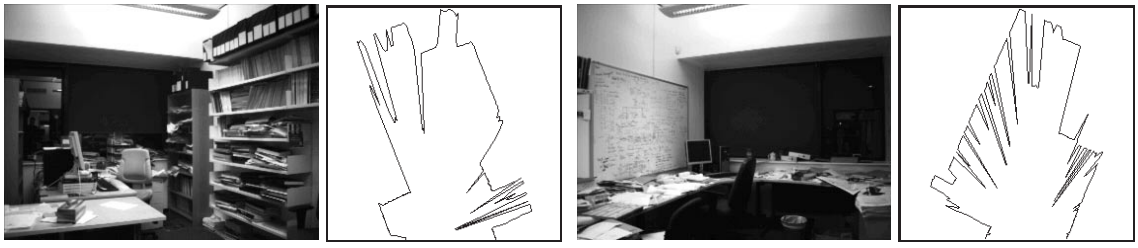
### *7.2. Experimental Procedure and Results*

Three days after the training data were collected, we performed a live experiment in the lab on the second floor in the same building. The experiment was conducted during the day with sunny weather. The part of the environment that was explored by the robot consisted of two offices (Nick's office and Jeremy's office), a corridor and a meeting room. The interiors of the rooms and the influence of illumination can be seen in the images in Figure 16(b).

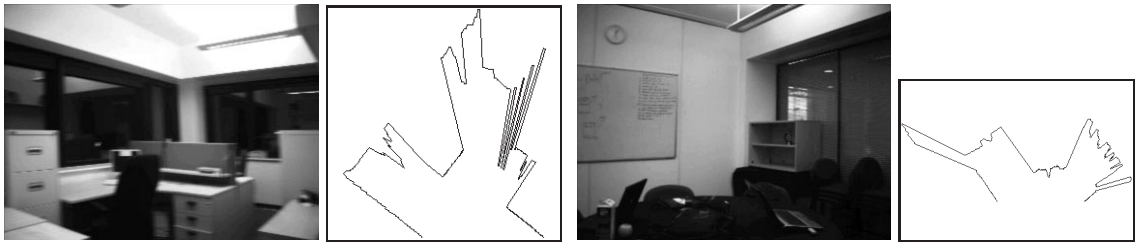
The SLAM system of the robot constructs a metric map and navigation graph. In this experiment, the task is to semantically label the navigation graph nodes and areas as the map is being built. The only knowledge given to the robot before the experiment consisted of the models of the three place classes: "office", "corridor" and "meeting room". As stated in Section 5, every time the robot created or revisited a node, the accumulated beliefs about the semantic category of the area were used to label the node and saved as a future prior. The label was also propagated to the whole area. We used detected doors to assign nodes to areas.

The whole experiment was videotaped and a video presenting the experimental setup, the test run and visualization of

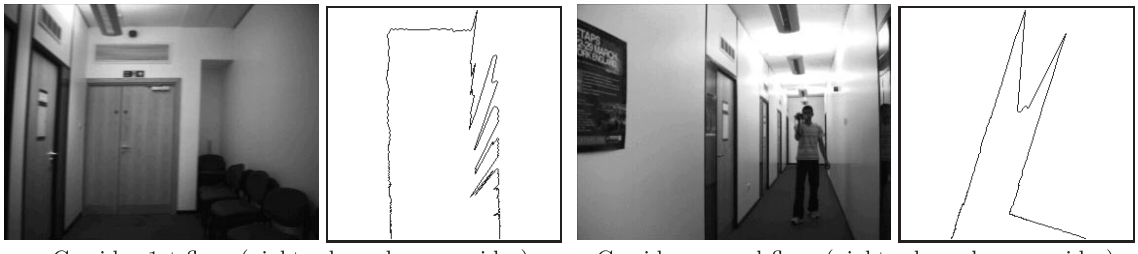
6. See http://www.cs.bham.ac.uk/research/projects/cosy/cast/


Aaron's office (night, place class: office)


Richrd's office (night, place class: office)

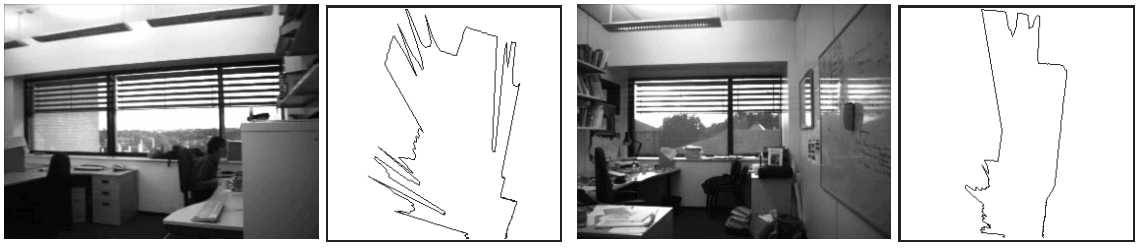

Robert's office (night, place class: office)


Meeting room (cloudy)

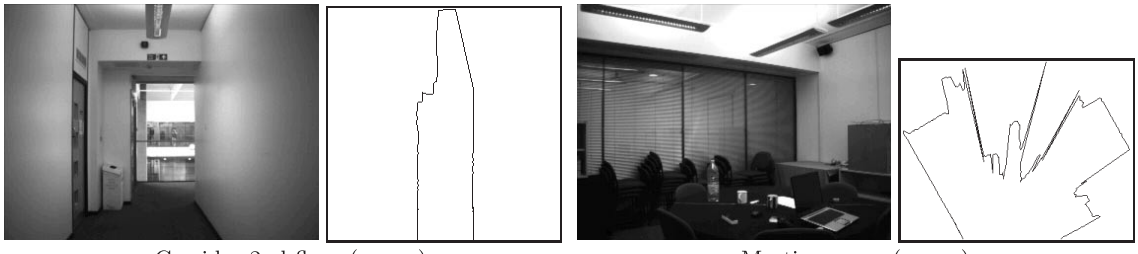

Corridor 1st floor (night, place class: corridor)


Corridor ground floor (night, place class: corridor)

(a) Samples from the data sequences used to train the models of place classes.


Nick's office (sunny)


Jeremy's office (sunny)


Corridor 2nd floor (sunny)


Meeting room (sunny)

(b) Samples acquired during the test run.

Fig. 16. Examples of images and laser scans (synchronized) taken from the data sequences used for training the models of place classes (a) and acquired during the test run (b) in each of the rooms considered during the experiment. The within-category variations for corridors and offices are illustrated as well as other types of variability observed for each place class (e.g. different illumination conditions, activity in the environment).

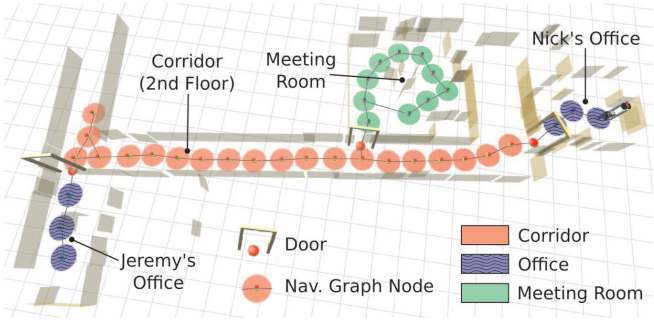



Fig. 17. Final map obtained after the test run. The navigation graph is overlaid on the metric map and the color of the circles around the graph nodes indicate the place class assigned to each area bounded by detected doors. The system correctly labeled all of the areas in the environment.

the obtained results can be found in Extension 2. The robot started in Nick's office, and was manually driven through the corridor to Jeremy's office. Then, it was taken to the meeting room where the autonomous exploration mode was turned on. The robot used a frontier-based algorithm based on Yamauchi (1997). Laser data was limited to 2 m distance in the exploration to make sure that the robot not just perceived how the environment looked but also covered it to build the navigation graph. After the meeting room was explored, the robot was manually driven back to Nick's office where the experiment finished. A video presenting visualization of the full test run is available in Extension 5. The labeling process was running online and the place classification was performed approximately at the rate of 5 times per second. The final semantic map build during the run is shown in Figure 17. We can see that the system correctly labeled all the areas in the environment.

The sensory data acquired during the test run are available as Extension 4. Moreover, a video presenting the sequence of images and laser scans is presented in Extension 6. The fact that the data were stored allowed for additional performance analysis of the multi-modal place classification system, similar to the one presented in Section 6. The results are displayed in Figure 18. When we look at the overall classification rate for all the data samples in the test sequence, we see that vision significantly outperformed laser in this experiment (66% versus 84%). Still, the performance of the system was boosted by an additional 8% compared with vision alone when the two modalities were integrated. The gain is even more apparent if we look at the detailed results for each of the classes (the first three charts in Figure 18). We see that the modalities achieved different performance, but also different error patterns, for each class. Clearly, the system based on laser range data is a very good corridor detector. On the other hand, vision was able to distinguish between the offices and the meeting room almost perfectly. Finally, the integrated system always achieved the performance of the more reliable modality and for two out of three classes outperformed the uni-modal systems. As can be seen in the video in Extensions 2 and 5, this provided stable performance for each of the classes and a robust base for the semantic labeling system.

## 8. Conclusions

In this paper we have addressed the problem of place classification and showed how it can be applied to semantic knowledge extraction in robotic systems. This is an important and challenging task, where multiple sensor modalities are necessary in order to achieve generality and robustness, and enable systems to work in realistic settings. To this end, we presented a new cue integration method able to combine multiple cues derived by a single modality, as well as cues obtained by multiple sensors. The method was thoroughly tested in off-line experiments on realistic data collected under varying conditions and as part of a real-time system running on a robotic platform. The results obtained using multiple visual cues alone, and combined with laser range features, clearly show the value of our approach. Finally, we showed that the system can successfully be applied for the space labeling problem where it can be used to augment the internal space representation with semantic place information. All of the data used in the paper are available as extensions to the paper and from the IDOL2 database (Luo et al. 2006).

In the future, we plan to extend this method and attack the scalability issue, with particular attention to indoor office environments. These are usually characterized by a large number of rooms with very similar characteristics; we expect that in such a domain our approach will be particularly effective. Another important aspect of place classification is the intrinsic dynamics in the sensory information: as rooms are used daily, furniture is moved around, objects are taken in and out of drawers and people appear. All of this affects the sensor inputs of places in time. We plan to combine our approach with incremental extensions of the SVM algorithm (Luo et al. 2007; Orabona et al. 2007) and to extend these methods from fully supervised to semi-supervised learning, so to obtain a system able to learn continuously from multiple sensors.

## Acknowledgments

Special thanks go to Sagar Behere for his great help with running the integrated system on the robotic platform, data acquisition and videotaping. This work was sponsored by the EU integrated projects FP6-004250-IP CoSy (A. Pronobis, O. Martínez Mozos, P. Jensfelt), ICT-215181-CogX (A. Pronobis and P. Jensfelt) and IST-027787 DIRAC (B. Caputo), and the Swedish Research Council contract 2005-3600-Complex (A. Pronobis). The support is gratefully acknowledged.

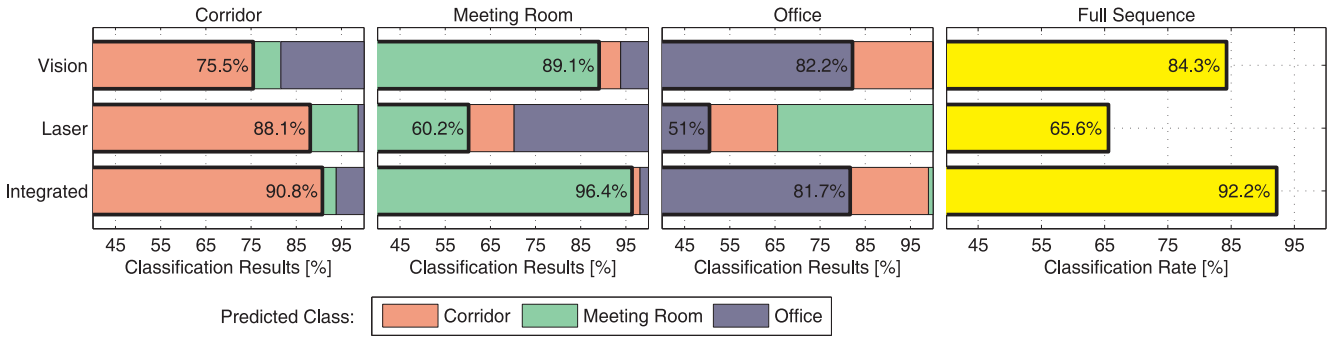



Fig. 18. Place classification results obtained on the dataset recorded during the test run. The first three bar charts show the results separately for each place class: "corridor", "meeting room" and "office". The charts show the percentage of the samples that were properly classified (most left bars marked with thick lines), but also how the misclassifications were distributed. The chart on the right presents the percentage of properly classified samples during the whole run. The two top rows give results for single modalities, while the bottom row shows results for the multi-modal system.

A preliminary version of part of the experimental evaluation reported in this work was presented in Pronobis et al. (2008).

## Appendix: Index to Multimedia Extensions

The multimedia extension page is found at http://www.ijrr.org

**Table of Multimedia Extensions**

| Extension | Type | Description |
|---|---|---|
| 1 | Video | The acquisition procedure of a typical data sequence in the IDOL2 database. |
| 2 | Video | The setup, procedure and visualization of the experiment with semantic space labeling based on multi-modal place classification. |
| 3 | Video | The process of acquiring data for training the models of places for the experiment with semantic space labeling. |
| 4 | Data | The dataset (sequences of images and laser scans) collected during the experiment with semantic labeling of space. |
| 5 | Video | Visualization of the complete test run and results obtained during the experiment with semantic space labeling. |
| 6 | Video | The complete sequence of images and laser scans acquired during the test run of the experiment with semantic space labeling. |

## References

Aloimonos, J. and Shulman, D. (1989). *Integration of Visual Modules: an Extension of the Marr Paradigm*. New York, Academic Press.

Althaus, P. and Christensen, H. I. (2003). Behaviour coordination in structured environments. *Advanced Robotics*, **17**(7): 657–674.

Andreasson, H., Treptow, A. and Duckett, T. (2005). Localization for mobile robots using panoramic vision, local features and particle filter. *Proceedings of the International Conference on Robotics and Automation (ICRA)*, Barcelona, Spain.

Barla, A., Odone, F. and Verri, A. (2003). Histogram intersection kernel for image classification. *Proceedings of the International Conference on Image Processing (ICIP)*, Barcelona, Spain.

Bay, H., Tuytelaars, T. and Van Gool, L. J. (2006). Surf: Speeded up robust features. *Proceedings of the 9th European Conference on Computer Vision (ECCV)*, Graz, Austria.

Blaer, P. and Allen, P. (2002). Topological mobile robot localization using fast vision techniques. *Proceedings of the International Conference on Robotics and Automation (ICRA)*, Washington, DC.

Bradley, D. M., Patel, R., Vandapel, N. and Thayer, S. M. (2005). Real-time image-based topological localization in large outdoor environments. *Proceedings of the International Conference on Intelligent Robots and Systems (IROS)*, August, Edmonton, AB, Canada.

Buschka, P. and Saffiotti, A. (2002). A virtual sensor for room detection. *Proceedings of the International Conference on Intelligent Robots and Systems (IROS)*, Lausanne, Switzerland.

Caputo, B. and Dorko, G. (2002). How to combine color and shape information for 3D object recognition: Kernels do the

trick. *Neural Information Processing Systems*, Vancouver, BC, Canada.

Chapelle, O., Haffner, P. and Vapnik, V. (1999). Support vector machines for histogram-based image classification. *Transactions on Neural Networks*, **10**(5): 1055–1064.

Clark, J. and Yuille, A. (1990). *Data Fusion for Sensory Information Processing Systems*. Dordrecht, Kluwer.

Cristianini, N. and Shawe-Taylor, J. (2000). *An Introduction to Support Vector Machines and Other Kernel-based Learning Methods*. Cambridge, Cambridge University Press.

Demšar, J. (2006). Statistical comparisons of classifiers over multiple data sets. *Journal of Machine Learning Research*, **7**: 1–30.

Douillard, B., Fox, D. and Ramos, F. (2007). A spatio-temporal probabilistic model for multi-sensor object recognition. *Proceedings of the International Conference on Intelligent Robots and Systems (IROS)*, San Diego, CA.

Duda, R., Hart, P. and Stork, D. (2001). *Pattern Classification*, 2nd edn. New York, Wiley.

Filliat, D. (2007). A visual bag of words method for interactive qualitative localization and mapping. *Proceedings of the International Conference on Robotics and Automation (ICRA)*, Rome, Italy.

Fraundorfer, F., Engels, C. and Nistér, D. (2007). Topological mapping, localization and navigation using image collections. *Proceedings of the International Conference on Intelligent Robots and Systems (IROS)*, October, San Diego, CA.

Freund, Y. and Schapire, R. E. (1995). A decision-theoretic generalization of on-line learning and an application to boosting. *Proceedings of the European Conference on Computational Learning Theory*, Barcelona, Spain.

Galindo, C., Saffiotti, A., Coradeschi, S., Buschka, P., Fernández-Madrigal, J. and González, J. (2005). Multi-hierarchical semantic maps for mobile robotics. *Proceedings of the International Conference on Intelligent Robots and Systems (IROS)*, Edmonton, AB, Canada.

Gaspar, J., Winters, N. and Santos-Victor, J. (2000). Vision-based navigation and environmental representations with an omni-directional camera. *Transactions on Robotics and Automation*, **16**(6): 890–898.

Koenig, S. and Simmons, R. G. (1998). Xavier: A robot navigation architecture based on partially observable Markov decision process models. *Artificial Intelligence Based Mobile Robotics: Case Studies of Successful Robot Systems*, Kortenkamp, D., Bonasso, R. and Murphy, R. (eds). Cambridge, MA, MIT Press, pp. 91–122.

Kuipers, B. (2006). An intellectual history of the spatial semantic hierarchy. *Robot and Cognitive Approaches to Spatial Mapping*. Berlin, Springer.

Kuipers, B. and Beeson, P. (2002). Bootstrap learning for place recognition. *Proceedings of the 18th National Conference on Artificial Intelligence (AAAI)*, Edmonton, AB, Canada.

Linde, O. and Lindeberg, T. (2004). Object recognition using composed receptive field histograms of higher dimensionality. *Proceedings of the International Conference on Pattern Recognition (ICPR)*, Cambridge, UK.

Lowe, D. G. (2004). Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, **60**(2): 91–110.

Luo, J., Pronobis, A., Caputo, B. and Jensfelt, P. (2006). *The IDOL2 Database*. Technical Report, KTH, CAS/CVAP. Available at http://www.cas.kth.se/IDOL/.

Luo, J., Pronobis, A., Caputo, B. and Jensfelt, P. (2007). Incremental learning for place recognition in dynamic environments. *Proceedings of the International Conference on Intelligent Robots and Systems (IROS)*, San Diego, CA.

Matas, J., Marik, R. and Kittler, J. (1995). On representation and matching of multi-coloured objects. *Proceedings of the International Conference on Computer Vision (ICCV)*, Boston, MA.

Menegatti, E., Zoccarato, M., Pagello, E. and Ishiguro, H. (2004). Image-based Monte-Carlo localisation with omnidirectional images. *Robotics and Autonomous Systems*, **48**(1): 17–30.

Mozos, O. M., Stachniss, C. and Burgard, W. (2005). Supervised learning of places from range data using AdaBoost. *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, Barcelona, Spain, pp. 1742–1747.

Mozos, O. M., Triebel, R., Jensfelt, P., Rottmann, A. and Burgard, W. (2007). Supervised semantic labeling of places using information extracted from sensor data. *Robotics and Autonomous Systems*, **55**(5): 391–402.

Murillo, A. C., Guerrero, J. J. and Sagues, C. (2007). Surf features for efficient robot localization with omnidirectional images. *Proceedings of the International Conference on Robotics and Automation (ICRA)*, Rome, Italy.

Nilsback, M. E. and Caputo, B. (2004). Cue integration through discriminative accumulation. *Proceedings of the Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, Washington, DC.

Orabona, F., Castellini, C., Caputo, B., Luo, J. and Sandini, G. (2007). Indoor place recognition using online independent support vector machines. *Proceedings of the British Machine Vision Conference (BMVC)*, Warwick, UK.

Platt, J. C. (1999). Fast training of support vector machines using sequential minimal optimization. *Advances in Kernel Methods: Support Vector Learning*, Schölkopf, B., Burges, C. and Smola, A. (eds). Cambridge, MA, MIT Press, pp. 185–208.

Poggio, T., Torre, V. and Koch, C. (1985). Computational vision and regularization theory. *Nature* **317**: 314–319.

Polikar, R. (2006). Ensemble based systems in decision making. *IEEE Circuits and Systems Magazine*, **6**: 21–45.

Posner, I., Schroeter, D. and Newman, P. M. (2007). Describing composite urban workspaces. *Proceedings of the Inter-*

*national Conference on Robotics and Automation (ICRA)*, Rome, Italy.

Pronobis, A. and Caputo, B. (2007). Confidence-based cue integration for visual place recognition. *Proceedings of the International Conference on Intelligent Robots and Systems (IROS)*, San Diego, CA.

Pronobis, A., Caputo, B., Jensfelt, P. and Christensen, H. I. (2006). A discriminative approach to robust visual place recognition. *Proceedings of the International Conference on Intelligent Robots and Systems (IROS)*, October, Beijing, China.

Pronobis, A., Mozos, O. M. and Caputo, B. (2008). SVM-based discriminative accumulation scheme for place recognition. *Proceedings of the International Conference on Robotics and Automation (ICRA)*, May, Pasadena, CA.

Rothganger, F., Lazebnik, S., Schmid, C. and Ponce, J. (2006). 3D object modeling and recognition using local affine-invariant image descriptors and multi-view spatial constraints. *International Journal of Computer Vision*, **66**(3): 231–259.

Rottmann, A., Mozos, O. M., Stachniss, C. and Burgard, W. (2005). Semantic place classification of indoor environments with mobile robots using boosting. *Proceedings of the 20th National Conference on Artificial Intelligence (AAAI)*, Pittsburgh, PA.

Se, S., Lowe, D. G. and Little, J. (2001). Vision-based mobile robot localization and mapping using scale-invariant features. *Proceedings of the International Conference on Robotics and Automation (ICRA)*, Seoul, Korea.

Siagian, C. and Itti, L. (2007). Biologically-inspired robotics vision monte-carlo localization in the outdoor environment. *Proceedings of the International Conference on Intelligent Robots and Systems (IROS)*, October, San Diego, CA.

Stachniss, C., Mozos, O. M. and Burgard, W. (2006). Speeding-up multi-robot exploration by considering semantic place information. *Proceedings of the International Conference on Robotics and Automation (ICRA)*, Orlando, FL.

Stachniss, C., Grisetti, G., Mozos, O. and Burgard, W. (2007). Efficiently learning metric and topological maps with autonomous service robots. *Information Technology*, **49**: 232–237.

Tamimi, H. and Zell, A. (2004). Vision based localization of mobile robots using kernel approaches. *Proceedings of the International Conference on Intelligent Robots and Systems (IROS)*, Sendai, Japan.

Tapus, A. and Siegwart, R. (2005). Incremental robot mapping with fingerprints of places. *Proceedings of the International Conference on Intelligent Robots and Systems (IROS)*, Edmonton, AB, Canada.

Tommasi, T., Orabona, F. and Caputo, B. (2008). Discriminative cue integration for medical image annotation. *Pattern Recognition Letters, Special Issue on IMageCLEF Med Benchmark Evaluation*, **29**(15): 1996–2002.

Topp, E. A. and Christensen, H. I. (2006). Topological modelling for human augmented mapping. *Proceedings of the International Conference on Intelligent Robots and Systems (IROS)*, October, Beijing, China.

Torralba, A. (2003). Contextual priming for object detection. *International Journal of Computer Vision*, **53**(2): 169–191.

Torralba, A. and Sinha, P. (2001). *Recognizing Indoor Scenes*. Technical Report 2001-015, AI Memo.

Torralba, A., Murphy, K. P., Freeman, W. T. and Rubin, M. A. (2003). Context-based vision system for place and object recognition. *Proceedings of the International Conference on Computer Vision (ICCV)*, Nice, France.

Triesch, J. and Eckes, C. (1998). Object recognition with multiple feature types. *Proceedings of the International Conference on Artificial Neural Networks (ICANN)*, Skövde, Sweden.

Ulrich, I. and Nourbakhsh, I. (2000). Appearance-based place recognition for topological localization. *Proceedings of the International Conference on Robotics and Automation (ICRA)*, San Francisco, CA.

Valgren, C. and Lilienthal, A. J. (2008). Incremental spectral clustering and seasons: Appearance-based localization in outdoor environments. *Proceedings of the International Conference on Robotics and Automation (ICRA)*, Pasadena, CA.

Wallraven, C., Caputo, B. and Graf, A. (2003). Recognition with local features: the kernel recipe. *Proceedings of the International Conference on Computer Vision (ICCV)*, Nice, France.

Weiss, C., Tamimi, H., Masselli, A. and Zell, A. (2007). A hybrid approach for vision-based outdoor robot localization using global and local image features. *Proceedings of the International Conference on Intelligent Robots and Systems (IROS)*, October, San Diego, CA.

Yamauchi, B. (1997). A frontier-based approach for autonomous exploration. *Proceedings of the International Symposium on Computational Intelligence in Robotics and Automation*, Monterey, CA.

Zender, H., Jensfelt, P., Mozos, O. M., Kruijff, G.-J. M. and Burgard, W. (2007). An integrated robotic system for spatial understanding and situated interaction in indoor environments. *Proceedings of the 22nd National Conference on Artificial Intelligence (AAAI)*, Vancouver, BC, Canada.

Zender, H., Mozos, O. M., Jensfelt, P., Kruijff, G.-J. M. and Burgard, W. (2008). Conceptual spatial representations for indoor mobile robots. *Robotics and Autonomous Systems*, **56**(6): 493–502.

Zivkovic, Z., Bakker, B. and Kröse, B. (2005). Hierarchical map building using visual landmarks and geometric constraints. *Proceedings of the International Conference on Intelligent Robots and Systems (IROS)*, Edmonton, AB, Canada.

# The More you Learn, the Less you Store: Memory-controlled Incremental SVM for Visual Place Recognition

Andrzej Pronobis [a,*], Luo Jie [b,c], Barbara Caputo [b],

[a] *CAS/CVAP, The Royal Institute of Technology (KTH), Stockholm, Sweden*
[b] *Idiap Research Institute, Martigny, Switzerland*
[c] *Swiss Federal Institute of Technology in Lausanne (EPFL), Switzerland*

**Abstract**

The capability to learn from experience is a key property for autonomous cognitive systems working in realistic settings. To this end, this paper presents an SVM--based algorithm, capable of learning model representations incrementally while keeping under control memory requirements. We combine an incremental extension of SVMs [45] with a method reducing the number of support vectors needed to build the decision function without any loss in performance [15] introducing a parameter which permits a user-set trade-off between performance and memory. The resulting algorithm is able to achieve the same recognition results as the original incremental method while reducing the memory growth. Our method is especially suited to work for autonomous systems in realistic settings. We present experiments on two common scenarios in this domain: adaptation in presence of dynamic changes and transfer of knowledge between two different autonomous agents, focusing in both cases on the problem of visual place recognition applied to mobile robot topological localization. Experiments in both scenarios clearly show the power of our approach.

*Key words:* incremental learning, knowledge transfer, support vector machines, place recognition, visual robot localization

* Corresponding author.
*Email addresses:* pronobis@csc.kth.se (Andrzej Pronobis), jluo@idiap.ch (Luo Jie), bcaputo@idiap.ch (Barbara Caputo).

## 1 Introduction

Many recent advances in fields such as computer vision and robotics have been driven by the ultimate goal of creating artificial cognitive systems able to perform human-like tasks. Several attempts have been made to create integrated cognitive architectures and implement them, for instance, on mobile robots [2,23,1,3]. The ability to learn and interpret complex sensory information based on the previous experience, inherently connected with cognition, has been recognized as crucial and vastly researched [43,41,34]. In most cases, the recognition systems used are trained offline, i.e. they are based on batch learning algorithms. However, in the real, dynamic world, learning cannot be a single act. It is simply not possible to create a static model which could explain all the variability observed over time. Continuous information acquisition and exchange, coupled with an ongoing learning process, is necessary to provide a cognitive system with a valid world representation.

In artificial autonomous agents constrained by limited resources (such as mobile robots), continuous learning must be performed in an incremental fashion. It is obviously not feasible to rebuild the internal model from scratch every time new information arrives, neither it is possible to store all the previously acquired data for that purpose. The model must be updated and the updating process must have certain properties. First, the knowledge representation must remain compact and free from redundancy to fit into the limited memory and maintain a fixed computational complexity. We call this property *controlled memory growth*. Second, in the continuous learning scenario, a model cannot grow forever even though new information is constantly arriving. Thus, the updating process should be able to gradually filter out unnecessary information. We call this property *forgetting capability*.

Discriminative methods have become widely popular for visual recognition, achieving impressive results on several applications [49,20,14]. Within discriminative classifiers, SVM techniques provide powerful tools for learning models with good generalization capabilities; in some domains like object and material categorization, SVM-based algorithms are state of the art [7,17]. This makes it worth it to investigate whether it is possible to perform continuous learning with this type of methods. Several incremental extensions of SVMs have been proposed in the machine learning community [13,8,45,36]. Between these methods, the approximate techniques [13,45] seem better suited for visual recognition because, at each incremental step, they discard non-informative training vectors, thus reducing the memory requirements. Other methods, such as [8,36], instead require to store in memory all the training data, eventually leading to a memory explosion; this makes them unfit for real-time autonomous systems.

This paper presents an SVM-based incremental method which performs like the batch algorithm while reducing the memory requirements. We combine an approximate technique for incremental SVM [45] with an exact method that reduces the number of support vectors needed to build the decision function without any loss in performance [15]. This results in an algorithm performing as the original incremental method with a reduction in the memory requirements. We then present an extension of the method for the exact simplification of the support vector solution [15]. We introduce a parameter that links the performance of an SVM to the amount of vectors that is possible to discard. This allows a user-set trade-off between performance and memory reduction.

We evaluate the suitability of our method for autonomous cognitive systems in two challenging scenarios: adaptation in presence of dynamic changes and transfer of knowledge between autonomous agents. In both cases, we concentrate on the problem of visual place recognition applied to mobile robot topological localization. The problem is important from the point of view of engineering cognitive systems, as it allows to tie semantics with space representations and provides solutions for typical problems with purely metric localization. However, it is also a challenging recognition problem as it requires processing of large amounts of high-dimensional visual information which is noisy and dynamic in nature. In this context, the memory and computational efficiency become one of the most important properties of the learning algorithm determining the design choice.

In our considerations, we first focus on the scenario in which the incremental learning is used to provide adaptability to different types of variations observed in real-world environments. In our previous work [40,38], we presented a purely appearance-based model able to cope with illumination and pose changes, and we showed experimentally that it could achieve satisfactory performances when considering short time intervals between the acquisition of the training and testing data. Nevertheless, a room's appearance is doomed to change dramatically over time because it is used: chairs are pushed around, objects are taken in/out of drawers, furniture and paintings are added, or changed, or re-arranged; and so forth. As it is not possible to predict a priori how a room is going to change, the only possible strategy is to update the representation over time, learning incrementally from the new data recorded during use.

As a second scenario, we consider the case when a robot, proficient in solving the place recognition task within a known environment, transfers its visual knowledge to another robotic platform with different characteristics, which is a tabula rasa. The ability to transfer knowledge between different domains enables humans to learn efficiently from small number of examples. This observation inspired robotics and machine learning researchers to search for algorithms able to exploit prior knowledge so to improve performance of artificial learners and speed up the learning process. To tackle this problem, it is neces-

sary an efficient way of exploiting the knowledge transferred from a different platform as well as updating the internal representation when new training data are available. The knowledge transfer scheme should be adaptive and privilege newest data so to prevent from accumulating outdated information. Finally, the solution obtained starting from a transferred model should gradually converge to the one learned from scratch, not only in terms of performance on a task but also of required resources (e.g. memory).

To achieve these goals, we used our memory-controlled incremental SVM and we evaluated its performance in terms of accuracy, memory growth, complexity and forgetting capability. We compare the results obtained by our method with those achieved by the batch algorithm and by two other incremental extensions of SVMs, one approximate (the fixed-partition incremental SVM, [45]) and one exact (online independent SVM, [36]). We evaluated the algorithms on a visual place recognition database acquired using two mobile robot platforms [40], which we extended with new data acquired 6 months later using the same hardware. Then, we confirmed the results on another database acquired in a different environment and using different hardware [39]. To test the adaptability of the recognition system, we performed topological localization experiments under realistic long-term variations. To test the knowledge transfer capabilities, we performed experiments in case of which visual knowledge captured in the SVM model was gradually exchanged between the two mobile robot platforms. The experiments clearly show the power of our approach in both scenarios, while illustrating the need for incremental solutions in artificial cognitive systems.

The rest of the paper is organized as follows: after a review of related work (Section 2), Section 3 gives our working definition of visual place recognition for robot localization. Section 4 reviews SVMs, it introduces the memory-controlled incremental SVM algorithm, which will constitute a building block of the adaptive place recognition system and a base for our knowledge transfer technique, and it briefly describes two other incremental extensions of SVMs against which we will benchmark our approach. Section 5 describes our experimental setup; Section 7 concentrates on the adaptation problem and presents experimental evaluation of the algorithms in this context. Finally, Section 8 gives details of our approach to the transfer of knowledge and shows its effectiveness with a set of experiments. The paper concludes with a summary and possible directions for future work.

## 2 Related Work

In the last years, the need for solutions to such problems as robustness to long-term dynamic variations or transfer of knowledge is more and more ac-

knowledged. In [41], the authors tried to deal with long-term visual variations in indoor environments by combining information acquired using two sensors of different characteristics. In [51], the problem of invariance to seasonal changes in appearance of an outdoor environment is addressed. Clearly, adaptability is a desirable property of a recognition system. At the same time, Thrun and Mitchell [48,33] studied the issue of exchanging knowledge related to different tasks in the context of artificial neural networks and argued for the importance of knowledge-transfer schemes for lifelong robot learning. Several attempts to solve the problem have also been made from the perspective of Reinforcement Learning, including the case of transferring learned skills between different RL agents [30,21].

The work conducted in the fields of cognitive robotics and vision stimulated the research in the machine learning community directed towards developing extensions for algorithms that were commonly used due to their superior performance but were missing the ability to be trained incrementally. As a result, methods such as Incremental PCA have been invented and successfully applied e.g. for mobile robot localization [4,11]. As it was already mentioned, several incremental extensions have been introduced also for Support Vector Machines [13,8,45]. Between these methods, the approximate techniques[13,45] seem better suited for visual recognition because, at each incremental step, they discard non-informative training vectors, thus reducing the memory requirements. Other methods, such as [8,36], or simple KNN-based solutions, instead require to store in memory all the training data, eventually leading to a memory explosion. This limits their usefulness for complex real-world problems involving continuous learning of visual patterns.

Despite the fact that the approximate incremental SVM extensions allow to reduce the amount of data stored during the learning process, there is no guarantee that the continuously updated model will not grow forever. Additionally, the results of experiments that can be found in the literature do not give a clear answer if it is possible to apply such methods for complex problems such as visual place recognition or transfer of visual knowledge.

## 3 Visual Place Recognition for Robot Localization

In this section, we give our working definition of visual place recognition, explaining how it can be applied to mobile robot topological localization. We define a place as a nameable segment of a real-world environment, uniquely identifiable because of its specific functionality and/or appearance. Examples of places, according to this definition, are a kitchen, an office, a corridor, and so forth. We adopt the appearance-based paradigm, and we assume that a realistic scene can be represented by a visual descriptor without any loss of dis-

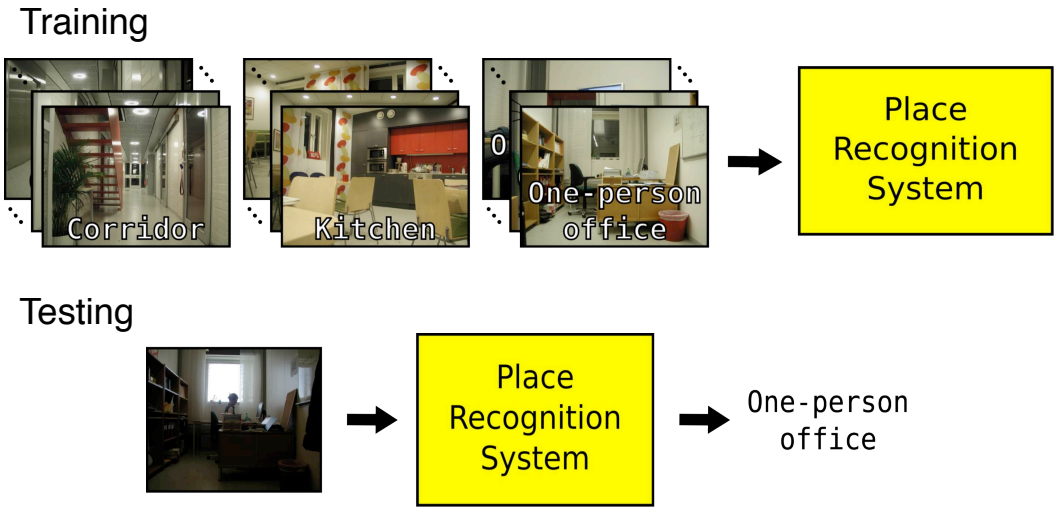



Fig. 1. A schematic representation of our visual place recognition system for robot localization.

criminative information. We consider a fully supervised, incremental learning scenario: we assume that, at each incremental step, every room is represented by a collection of images which capture its visual appearance under different viewpoints, at a fixed time and illumination setting. During testing, the algorithm is presented with images of the same rooms, acquired under similar viewpoints but possibly under different illumination conditions and after some time, with a time range going from some minutes to several months. The goal is to recognize correctly each single image seen by the system. Fig. 1 illustrates the approach.

A typical application for an indoor place recognition system is topological robot localization. The localization problem is vastly researched. This resulted, over the years, in a broad range of approaches spanning from purely metric [19,12,54,16], to topological [50,31,41], and hybrid [47,6]. Traditionally, sonar and/or laser have been the sensory modalities of choice [35,31]. Yet, the inability to capture many aspects of complex realistic environments leads to the problem of perceptual aliasing [24], and greatly limits the usefulness of such methods for semantic mapping. Recent advances in vision have made this modality emerge as a natural and viable solution for localization problems. Vision provides richer sensory input allowing for better discrimination. It opens new possibilities for building cognitive systems, actively relying on semantic context. Not unimportant is the cost effectiveness, portability and popularity of visual sensors. As a result, despite the complexity of the problem, this research line is attracting more and more attention, and several methods have been proposed using vision alone [42,50,49,38,44], or combined with more traditional range sensors [22,46,41].

Our visual place recognition system uses SVM-based discriminative place models trained on global and local image features. These features are described in

details in Section 5. The classification algorithm is introduced in Section 4. In our experiments, we always used only a single image as input for the recognition system. This makes the recognition problem harder, but also it makes it possible to perform global localization where no prior knowledge about the position is available (e.g. in case of the kidnapped robot problem). Spatial or temporal filtering can be used together with the presented method to enhance performance.

## 4 Memory-controlled Incremental SVM

This section describes our algorithmic approach to incremental learning of visual place models. We propose a fully supervised, SVM-based method with controlled memory growth that tends to privilege newest information over older data. This leads to a system able to adapt over time to the natural changes of a real-world setting, while maintaining a limited memory size and computational complexity.

The rest of this section describes the basic principles of Support Vector Machines (Section 4.1), a popular incremental extension of the basic algorithm (Section 4.2), our memory-controlled version of incremental SVM (Section 4.3) and an exact method based on a similar intuition (Section 4.4), with which we will compare our approach.

### *4.1 SVM: the batch algorithm*

Consider the problem of separating the set of training data $(\boldsymbol{x}_1, y_1), \ldots (\boldsymbol{x}_m, y_m)$ into two classes, where $\boldsymbol{x}_i \in \Re^N$ is a feature vector and $y_i \in \{-1, +1\}$ its class label (for multi-class extensions, we refer the reader to [10,52]). If we assume that the two classes can be linearly separated when mapped to some higher dimensional Hilbert space $\mathcal{H}$ by $\boldsymbol{x} \rightarrow \Phi(\boldsymbol{x}) \in \mathcal{H}$ (see [10,52] for solutions to non-separable cases), the optimal hyperplane is the one which has maximum distance to the closest points in the training set, resulting in a classification function:

$$f(x) = \text{sgn}\left(\sum_{i=1}^{m} \alpha_i y_i K(\boldsymbol{x}_i, \boldsymbol{x}) + b\right), \tag{1}$$

where $K(\boldsymbol{x}, \boldsymbol{y}) = \Phi(\boldsymbol{x}) \cdot \Phi(\boldsymbol{y})$ is the kernel function. Most of the $\alpha_i$'s take the value of zero; $\boldsymbol{x}_i$ with nonzero $\alpha_i$ are the Support Vectors (SV). Different kernel functions correspond to different similarity measures. Choosing a suitable kernel can therefore have a strong impact on the performance of the classifier. Based on results reported in the literature [40], here we used the two following kernels:

- The $\chi^2$ kernel [5] for histogram-like global descriptors:

$$K(\boldsymbol{x}, \boldsymbol{y}) = \exp\{-\gamma\chi^2(\boldsymbol{x}, \boldsymbol{y})\}, \qquad \chi^2(\boldsymbol{x}, \boldsymbol{y}) = \sum_{i=1}^{N} \frac{(x_i - y_i)^2}{x_i + y_i};$$

- The matching kernel [53] for local features:

$$K(\boldsymbol{L}_h, \boldsymbol{L}_k) = \frac{1}{n_h} \sum_{j_h=1}^{n_h} \max_{j_k=1,\ldots,n_k} \left\{ K_l(\boldsymbol{L}_h^{j_h}, \boldsymbol{L}_k^{j_k}) \right\},$$

  where $\boldsymbol{L}_h, \boldsymbol{L}_k$ are local feature sets and $\boldsymbol{L}_h^{j_h}, \boldsymbol{L}_k^{j_k}$ are two single local features. The sum is always calculated over the smaller set of local features and only some fixed amount of best matches is considered in order to exclude outliers. The local feature similarity kernel $K_l$ can be any Mercer kernel. We used the RBF kernel based on the Euclidean distance for the SIFT [27] features:

$$K_l(\boldsymbol{L}_h^{j_h}, \boldsymbol{L}_k^{j_k}) = \exp\left\{-\gamma||\boldsymbol{L}_h^{j_h} - \boldsymbol{L}_k^{j_k}||^2\right\}.$$

### *4.2 SVM: an Incremental Extension*

Among the incremental SVM extensions proposed so far [45,13,8], approximate methods seem to be the most suitable for visual recognition, because they discard a significant amount of the training data at each incremental step. Exact methods instead need to retain all training samples in order to preserve the convexity of the solution at each incremental step. As a consequence, they require huge amounts of memory when employed in realistic, continuous learning scenario as the one we consider here. Approximate methods avoid this problem by sacrificing the guaranteed optimality of the solution. Still, several studies showed that they generally achieve performances very similar to those obtained by an SVM trained on the complete data set (see [13] and references therein), because at each incremental step the algorithm remembers the essential class boundary information regarding the data seen so far (in form of support vectors). This information contributes properly to generate the classifier at the next iteration.

Once a new batch of data is loaded into memory, there are different possibilities for performing the update of the current model, which might discard a part of the new data according to some fixed criteria [13,45]. For all the techniques, at each step only the learned model from the data previously seen (preserved in form of SV) is kept in memory. In this paper we will consider the fixed-partition method [45]. Here the training data set is partitioned in batches of some size $k$:

$$\boldsymbol{T} = \{\boldsymbol{T}_1, \boldsymbol{T}_2, \ldots \boldsymbol{T}_n\},$$

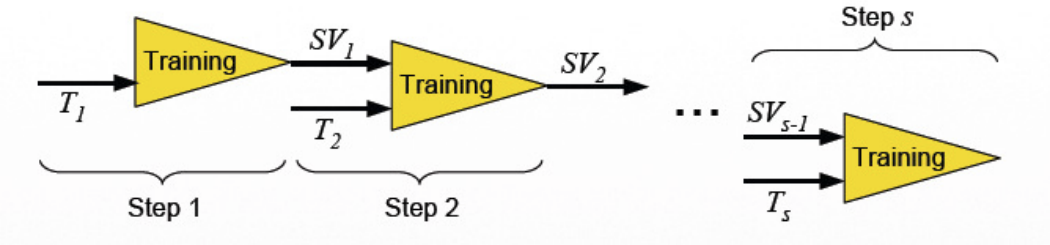



Fig. 2. The fixed-partition incremental SVM algorithm.

with $\boldsymbol{T}_i = \{(\boldsymbol{x}_j^i, y_j^i)\}_{j=1}^k$. At the first step, the model is trained on the first batch of data $\boldsymbol{T}_1$, obtaining a classification function

$$f_1(\boldsymbol{x}) = \text{sgn}\left(\sum_{i=1}^{m_1} \alpha_i^1 y_i^1 K(\boldsymbol{x}_i^1, \boldsymbol{x}) + b^1\right). \tag{2}$$

At the second step, a new batch of data is loaded into memory and added to the current set of support vectors; then, the *new* training set becomes

$$\boldsymbol{T}_2^{inc} = \{\boldsymbol{T}_2 \cup \boldsymbol{SV}_1\}, \quad \boldsymbol{SV}_1 = \{(\boldsymbol{x}_i^1, y_i^1)\}_{i=1}^{m_1},$$

where $\boldsymbol{SV}_1$ are the support vectors learned at the first step. The new classification function will be:

$$f_2(\boldsymbol{x}) = \text{sgn}\left(\sum_{i=1}^{m_2} \alpha_i^2 y_i^2 K(\boldsymbol{x}_i^2, \boldsymbol{x}) + b^2\right).$$

Thus, as new batches of data points are loaded into memory, the existing support vector model is updated, so to generate the classifier at that incremental step. The method is illustrated in Fig. 2. Note that this incremental method can be seen as an approximation of the chunking technique used for training SVM [10,52]. Indeed, the chunking algorithm is an exact decomposition which iterates through the training set to select the support vectors. The fixed-partition incremental method instead scan through the training data just once, and once discarded, does not consider them anymore. The fixed-partition incremental algorithm has been tested on several benchmark databases commonly used in the machine learning community [13], obtaining good performances comparable to the batch algorithm and other approximate methods. An open issue is that in principle there is no limitation to the memory growth. Indeed, several experimental evaluations show that, while approximate methods generally achieve classification performances equivalent to those of batch SVM, the number of SV tends to grow proportionally to the number of incremental steps (see [13] and references therein).

*4.3 Memory-controlled Incremental SVM*

The core idea of the memory-controlled incremental SVM is that the set of support vectors $\boldsymbol{X} = \{\boldsymbol{x}_i\}_{i=1}^m$ in Eq. (1) is not guaranteed to be linearly independent. Based on this observation, it is possible to reduce the number of support vectors of a trained classifier, eliminating those which can be expressed as a linear combination of the others in the feature space, as proposed in [15] for reducing the complexity of the SVM solution. By updating the weights accordingly, it is ensured that the decision function is exactly the same as the original one. More specifically, let us suppose that the first $r$ support vectors are linearly independent, and the remaining $m - r$ depend linearly on those in the feature space: $\forall j = r+1, \ldots m$, $\boldsymbol{x}_j \in span\{\boldsymbol{x}_i\}_{i=1}^r$. Then it holds

$$K(\boldsymbol{x}, \boldsymbol{x}_j) = \sum_{i=1}^{r} c_{ij} K(\boldsymbol{x}, \boldsymbol{x}_i), \tag{3}$$

and the classification function (1) can be rewritten as

$$f(\boldsymbol{x}) = \operatorname{sgn}\left(\sum_{i=1}^{r} \alpha_i y_i K(\boldsymbol{x}, \boldsymbol{x}_i) + \sum_{j=r+1}^{m} \alpha_j y_j \sum_{i=1}^{r} c_{ij} K(\boldsymbol{x}, \boldsymbol{x}_i) + b\right). \tag{4}$$

If we define the coefficients $\gamma_{ij}$ such that $\alpha_j y_j c_{ij} = \alpha_i y_i \gamma_{ij}$ and $\gamma_i = \sum_{j=r+1}^m \gamma_{ij}$, then Eq. (4) can be written as

$$f(\boldsymbol{x}) = \operatorname{sgn}\left(\sum_{i=1}^{r} \alpha_i y_i K(\boldsymbol{x}, \boldsymbol{x}_i) + \sum_{i=1}^{r} \alpha_i y_i \sum_{j=r+1}^{m} \gamma_{ij} K(\boldsymbol{x}, \boldsymbol{x}_i) + b\right)$$

$$= \operatorname{sgn}\left(\sum_{i=1}^{r} \alpha_i (1+\gamma_i) y_i K(\boldsymbol{x}, \boldsymbol{x}_i) + b\right) = \operatorname{sgn}\left(\sum_{i=1}^{r} \widehat{\alpha}_i y_i K(\boldsymbol{x}, \boldsymbol{x}_i) + b\right), \tag{5}$$

where

$$\widehat{\alpha}_i = \alpha_i (1+\gamma_i) = \alpha_i \left(1 + \sum_{j=r+1}^{m} \frac{\alpha_j y_j c_{ij}}{\alpha_i y_i}\right).$$

The $\alpha_i$ coefficients can be pre-multiplied by the class labels $\alpha_i' = \alpha_i y_i$ which results in a simple equation that can be used to obtain the weights of the reduced classifier:

$$\widehat{\alpha}_i' = \begin{cases} \alpha_i' + \sum_{j=r+1}^{m} \alpha_j' c_{ij} & \text{for } i = 1, 2, \ldots, r \\ 0 & \text{for } i = r+1, r+2, \ldots, m. \end{cases} \tag{6}$$

Thus, the resulting classification function (Eq. (5)) requires now $m - r$ less kernel evaluations than the original one.

The linearly independent subset of the support vectors as well as the coefficients $c_{ij}$ can be found by applying methods from linear algebra to the support

vector matrix given by

$$\boldsymbol{K} = \begin{bmatrix} K(\boldsymbol{x}_1, \boldsymbol{x}_1) & \cdots & K(\boldsymbol{x}_1, \boldsymbol{x}_m) \\ \vdots & \ddots & \vdots \\ K(\boldsymbol{x}_m, \boldsymbol{x}_1) & \cdots & K(\boldsymbol{x}_m, \boldsymbol{x}_m) \end{bmatrix}, \tag{7}$$

We employ the QR factorization with column pivoting [18] for this purpose. The QR factorization with column pivoting algorithm is a widely used method for selecting the independent columns of a matrix. The algorithm allows to reveal the numerical rank of the matrix with respect to a parameter $\tau$, which acts as a threshold in defining the condition of linear dependence. Additionally, it performs a permutation of the columns of the matrix so that they are ordered according to the degree of their relative linear independence. Consequently, if for a given value of $\tau$ the rank of the matrix is $r$, then the linearly independent columns will occupy the first $r$ positions.

The QR factorization with column pivoting of the matrix $\boldsymbol{K} \in \Re^{m \times m}$ is given by

$$\boldsymbol{K}\boldsymbol{\Pi} = \boldsymbol{Q}\boldsymbol{R}, \tag{8}$$

where $\boldsymbol{\Pi} \in \Re^{m \times m}$ is a permutation matrix, $\boldsymbol{Q} \in \Re^{m \times m}$ is orthogonal, and $\boldsymbol{R} \in \Re^{m \times m}$ is upper triangular. If we assume that the rank of the matrix $K$ with respect to the parameter $\tau$ equals $r$, then the matrices can be decomposed as follows:

$$\begin{bmatrix} \boldsymbol{K}_1 & \boldsymbol{K}_2 \end{bmatrix} = \begin{bmatrix} \boldsymbol{Q}_1 & \boldsymbol{Q}_2 \end{bmatrix} \begin{bmatrix} \boldsymbol{R}_{11} & \boldsymbol{R}_{12} \\ \boldsymbol{0} & \boldsymbol{R}_{22} \end{bmatrix}, \tag{9}$$

where the columns of $\boldsymbol{K}_1 \in \Re^{m \times r}$ create a linearly independent set, the columns of $\boldsymbol{K}_2 \in \Re^{m \times m-r}$ may be expressed as a linear combination of the columns of $\boldsymbol{K}_1$, $\boldsymbol{Q}_1 \in \Re^{m \times r}$, $\boldsymbol{Q}_2 \in \Re^{m \times m-r}$, $\boldsymbol{R}_{11} \in \Re^{r \times r}$, $\boldsymbol{R}_{12} \in \Re^{r \times m-r}$, $\boldsymbol{R}_{22} \in \Re^{m-r \times m-r}$.

The products of the QR factorization can be used to obtain the coefficients $c_{ij}$ as follows

$$\boldsymbol{C} = \begin{bmatrix} c_{1,r+1} & \cdots & c_{1,m} \\ \vdots & \ddots & \vdots \\ c_{r,r+1} & \cdots & c_{r,m} \end{bmatrix} = \boldsymbol{R}_{11}^{-1}\boldsymbol{Q}_1^T\boldsymbol{K}_2. \tag{10}$$

The coefficients together with the permutation matrix $\boldsymbol{\Pi} \in \Re^{m \times m}$ and the number of the linearly independent support vectors $r$ are sufficient to obtain the reduced solution. Using matrix notation, Eq. (6) can be expressed as follows

$$\begin{cases} \widehat{\boldsymbol{\alpha}}'_1 = \boldsymbol{\alpha}'_1 + \boldsymbol{R}_{11}^{-1}\boldsymbol{Q}_1^T\boldsymbol{K}_2\boldsymbol{\alpha}'_2 \\ \widehat{\boldsymbol{\alpha}}'_2 = \boldsymbol{0} \end{cases} \tag{11}$$

The rank $r$ of the matrix $\boldsymbol{K}$ can be estimated by thresholding $\|\boldsymbol{R}_{22}\|_2$ with the value of the parameter $\tau$. This means that, in practice, the choice of the $\tau$ value determines the number of linearly independent support vectors retained by the algorithm. For instance, by choosing a value of $\tau$ of 0.1 one will select a number of linearly independent support vectors smaller than by choosing a $\tau$ value of 0.01. This has two concrete effects on the algorithm:

(1) As the value of $\tau$ increases, the number of support vectors decreases. This means that, by tuning $\tau$, it is possible to reduce the memory requirements and to increase speed during classification;
(2) At the same time, as $\tau$ increases, Eq. (5) will become more and more an approximation of the exact solution, because we are considering as linearly dependent vectors that are not. Therefore, we are not able to preserve fully their informative content. Still, we don't lose all the information carried by the discarded support vector $\mathbf{x}_j$, as its weight $\alpha_j$ is used to compute the updated value of the weights $\widehat{\alpha_i}$ for the remaining support vectors. This should result in a graceful decrease of classification performance compared to the optimal solution.

We propose to combine this model simplification with the fixed-partition incremental algorithm, adding the reduction process at each incremental step. We call the new algorithm memory-controlled incremental SVM. It can be illustrated as follows:

(1) **Train**. The algorithm receives the first batch of data $\boldsymbol{T}_1$. It trains an SVM and obtains a set of support vectors $\boldsymbol{SV}_1$.
(2) **Find linearly dependent SVs**. The algorithm finds permutation of $\boldsymbol{SV}_1$ that orders the SVs according to the degree of their linear independence.
(3) **Find** $\tau$. The algorithm searches for the value of $\tau$, $\tau^\star$, that satisfies certain requirements regarding the number of support vectors or estimated performance of the classifier.
(4) **Reduce**. The algorithm computes the reduced solution determined by the chosen $\tau^\star$. After this step, the reduced model contains a subset of the original SVs, $\widehat{\boldsymbol{SV}}_1 = red(\boldsymbol{SV}_1)$, and can be used to classify test data.
(5) **Retrain**. As the new batch of data $\boldsymbol{T}_2$ arrives, step (1) is repeated using as training vectors $\widehat{\boldsymbol{T}}_2^{inc} = \{\boldsymbol{T}_2 \cup \widehat{\boldsymbol{SV}}_1\}$.

For applications that require speed and/or have limited memory requirements, at step (3) of the algorithm, one can tune $\tau$ so to obtain at each incremental step a predefined maximum number of stored SV. For applications where accuracy is more relevant, one can estimate at each incremental step the $\tau$ corresponding to a pre-defined maximum decrease in performance. This can be done on the batch of data $\boldsymbol{T}_i$ at each step, dividing $\boldsymbol{T}_i$ in two subsets and training on one and testing on the other or by applying the leave-one-

out strategy. We denote with the symbol $\Theta$ the percentage of the original classification rate that is guaranteed to be preserved after the reduction in this case.

In order to apply the method to multi-class problems, we used the one-vs-one multi-class extension. In a set of preliminary experiments comparing the one-vs-one and one-vs-all algorithms, we did not observe significant differences in the behavior of both methods (for further details, we refer the reader to [37]). The one-vs-one algorithm, given $M$ classes, trains $M(M-1)/2$ two-class SVMs, one for each pair of classes. In case of the place recognition experiments, this method obtained smaller training times due to large number of training samples and relatively small number of classes.

### *4.4 Online Independent Incremental SVM*

The idea to exploit the linear independence in the feature space has also been implemented in an online extension of SVMs, called Online Independent Support Vector Machine (OISVM, [36]). OISVM selects incrementally basis vectors that are used to build the solution of the SVM training problem, based upon linear independence in the feature space. Vectors that are linearly dependent on already stored ones are rejected. An incremental minimization algorithm is employed to find the new minimum of the cost function. This approach reduces considerably the complexity of the solution and therefore the testing time. As OISVM is an exact method, it requires to store all data acquired by the system during its whole life span for the update of the cost function. In many cases (e.g. in case of place recognition), the data samples are multi-dimensional and require a substantial amount of storage. Additionally, the learning algorithm needs to build a gram matrix the size of which is quadratic in the number of training samples. This leads inevitably to a memory explosion when the number of incremental steps grows, as we will show experimentally. Through its heuristics, the memory-controlled algorithm allows to decrease the number of training data samples at each incremental step and thus reduce the memory consumption.

## 5 Experimental Setup

This section describes our experimental setup. We first describe the IDOL2 and COLD-Freiburg databases, on which we will run all the experiments reported in this paper (Sections 5.1 and 5.2), then we briefly describe the feature representations used in the experiments (Section 5.3). Finally, we discuss


Fig. 3. Robot platforms employed in the experiments with the IDOL2 database and images illustrating the appearance of the five rooms from the robots' the point of view.

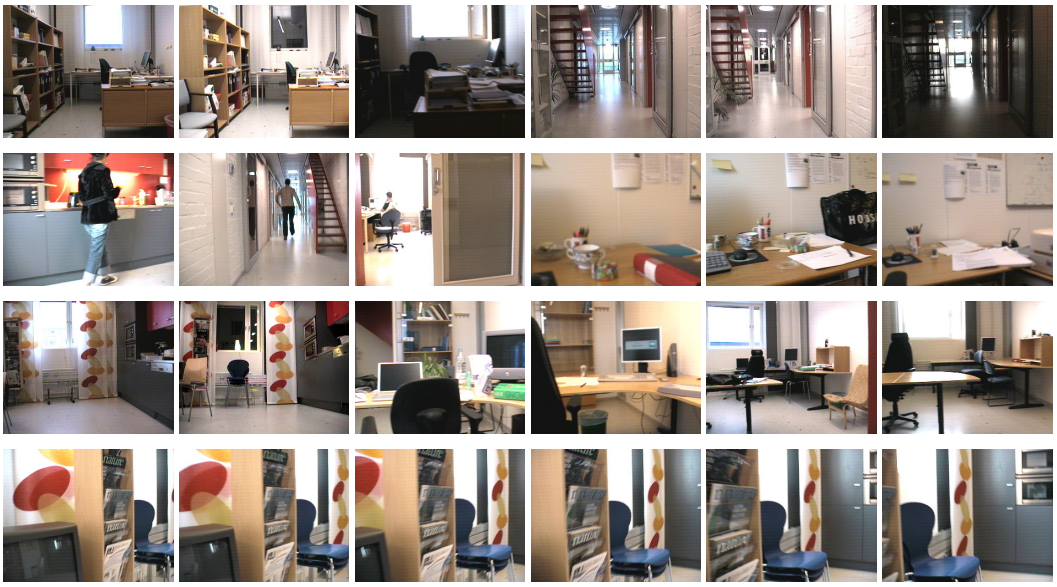


Fig. 4. Sample images illustrating the variations captured in the IDOL2 database. Images in the top row show the variability introduced by changes in illumination for two rooms. The second and third rows show people appearing in the environment (first three images, second row) as well as the influence of people's activity including some larger variations which happened over a time span of 6 months. Finally, the bottom row illustrates the changes in viewpoint observed for a series of images acquired one after another in 1.2 second.

the performance evaluation measure and parameter selection method (Section 5.4).

### *5.1 The IDOL2 Database*

The IDOL2 (Image Database for rObot Localization 2, [29]) database contains 24 image sequences acquired by a perspective camera, mounted on two mobile robot platforms. Both mobile robot platforms, the PeopleBot Minnie

and the PowerBot Dumbo, are equipped with cameras. On Minnie the camera is located 98cm above the floor, whereas on Dumbo its height is 36$cm$. Fig. 3 shows both robots and some sample images from the database acquired by the robots from very close viewpoints, illustrating the difference in visual content. These images were acquired under the same illumination conditions and within short time spans.

The robots were manually driven through an indoor laboratory environment and the images were acquired at a rate of 5fps. Each image sequence consists of 800-1100 frames automatically labeled with one of five different classes (Printer Area [PA], CoRridor [CR], KiTchen [KT], Two-persons Office [TO], and One-person Office [OO]). The labeling is based on the camera's position given by the laser-based localization system proposed in [16]. The acquisition procedure was repeated several times to capture the changes in illumination and varying weather conditions (sunny, cloudy, and night). Also, special care was taken to capture people's activities, change of location for objects and for furniture; for part of the environment (two-persons office) we were able to record a significant change in decoration which occurred over a time span of 6 months. Fig. 4 shows some sample images from the database, illustrating these variations. It is important to note that each single sequence captures the appearance of the considered experimental environment under stable illumination settings and during the short span of time that is required to drive the robot manually around the environment.

The 24 image sequences are divided as follows: for each robot platform and for each type of illumination conditions (cloudy, sunny, night), there are four sequences recorded. Of these four sequences, the first two were acquired six months before the last two. This means that, for every robot we always have subsets of sequences acquired under similar conditions and close in time, as well as subsets acquired under different conditions and distant in time. This makes the database useful for several types of experiments. It is important to note that, even for the sequences acquired within a short time span, variations still exist from everyday activities and viewpoint differences during acquisition. For further details, we refer the reader to [29].

### *5.2 The COLD-Freiburg Database*

The COLD-Freiburg database is a collection of image sequences acquired at the Autonomous Intelligent System Laboratory at the University of Freiburg and constitutes a part of the COsy Localization Database (COLD, [39]). The acquisition procedure of the COLD-Freiburg database was similar to that of the IDOL2 database. Image sequences were acquired using a mobile robot platform, under several illumination conditions (sunny, cloudy, night) and across

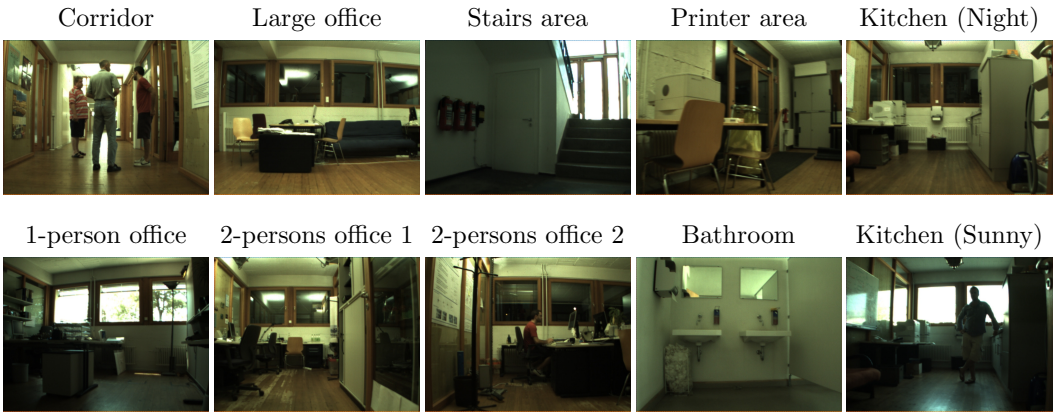



Fig. 5. Sample images from the COLD-Freiburg database illustrating the rooms in which acquisition was performed and different types of captured variability introduced by human activity and changes in illumination.

several days. As in case of IDOL2, special care was taken to capture people's activities and change of location of objects and furniture. However, the acquisition was performed using both perspective and omnidirectional cameras, in several parts of a different environment and using different hardware. For further details, we refer the reader to [39].

For our experiments, we employed only the perspective images and we selected 6 different extended sequences from the database. The extended sequences were acquired in a larger section of the environment consisting of 9 rooms of different functionality: a corridor, a printer area, a kitchen, a large office, 2 two-persons offices, a one-person office, a bathroom and a stairs area. The sequences contained on average 2547 frames. The 6 sequences were selected to mimic the organization of the IDOL2 database. For each illumination setting, we chose 2 sequences acquired under similar conditions and close in time.

### *5.3 Image Descriptors*

Two visual descriptors, global and local, were employed during our experiments. We used Composed Receptive Field Histograms (CRFH, [26]) as global features. CRFHs are a multi-dimensional statistical representation of the occurrence of responses of several image descriptors applied to the image. This idea is illustrated in Fig. 6. Each dimension corresponds to one descriptor and the cells of the histogram count the pixels sharing similar responses of all descriptors. This approach allows to capture various properties of the image as well as relations that occur between them. Multi-dimensional histograms can be extremely memory consuming and computationally expensive if the number of dimensions grows. In [26], Linde and Lindeberg suggest to exploit the fact that most of the cells are usually empty, and to store only those that are

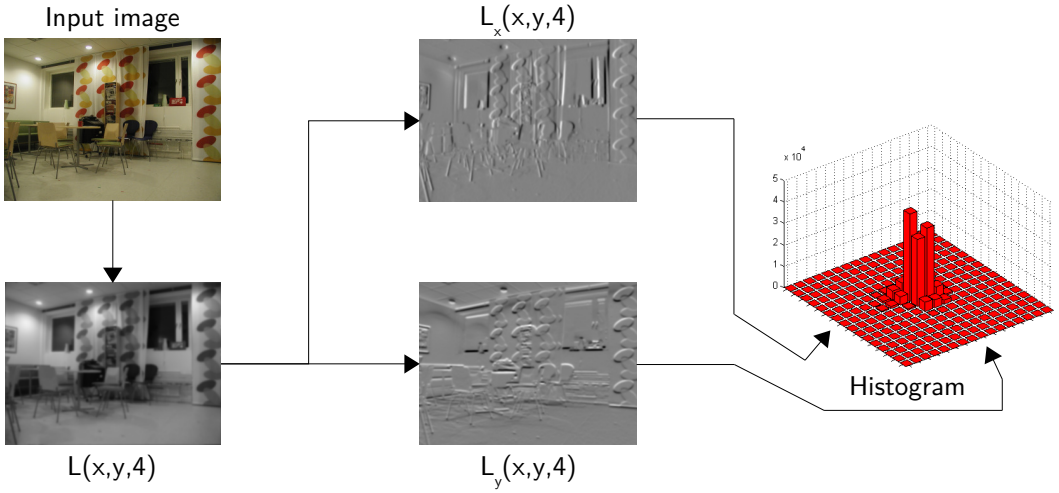



Fig. 6. The process of generating multi-dimensional receptive field histograms using the first-order derivatives computed at the scale $t = 4$ and the number of bins per dimension set to 16.

Fig. 7. Examples of images marked with interest points detected using the Harris-Laplace detector. The radius of the circles illustrate the scale at which the points were detected.

non-zero. This representation allows not only to reduce the amount of memory required, but also to perform operations such as histogram accumulation and comparison efficiently.

The idea behind local features is to represent the appearance of an image only around a set of characteristic points known as the interest points. The similarity between two images is then measured by solving the correspondence problem. Local features are known to be robust to occlusions and viewpoint changes, as the absence of some interest points does not affect the features extracted from other local patches. The process of local feature extraction consists of two stages: *interest point detection* and *description*. The interest point detector identifies a set of characteristic points in the image that could be re-detected even in spite of various transformations (e.g. rotation and scaling) and variations in illumination conditions. The role of the descriptor is to extract robust features from the local patches located at the detected points.

In this paper, we used the scale, rotation, and translation invariant Harris-Laplace detector [32] and the SIFT descriptor [28]. Fig. 7 shows two examples of interest point detected on images of indoor environments.

### *5.4 Parameter Selection and Performance Evaluation*

For all experiments, the kernel parameter and the SVM cost parameter $C$ were determined via cross validation, separately for each database. Then, the obtained values were used as constants for all the incremental learning experiments. For all experiments, we used the implementation of SVM provided by the *libsvm* library [9].

Since the employed datasets are unbalanced (e.g. in case of the IDOL2 database there are on average 443 samples for CR, 114 for 1pO, 129 for 2pO, 133 for KT and 135 for PR), as a measure of performance for the reported results and parameter selection, we used the average of classification rates obtained separately for each actual class. For each single experiment, the percentage of properly classified samples was first calculated separately for each room and then averaged with equal weights independently of the number of samples acquired in the room. This allowed to eliminate the influence that large classes could have on the performance score.

In our experiments, we observed a few percent improvement of the final results when a performance measure that is not invariant to unbalanced classes was used. This was caused by very good performance of the system for the corridor class. The was visually distinct from the other classes and was represented by the largest number of samples. As a result, in our experiments, the measure was used mainly to compensate for the influence of the corridor class.

## 6 Experiments on Support Vector Reduction

To begin with, we run some experiments to evaluate the behavior of the support vector reduction algorithm described in Section 4.3. We used two sequences from the IDOL2 database [29], one as train set and the other as test set. We chose CRFH as an image descriptor, and trained SVMs with four different types of kernels: linear kernel, RBF kernel, $\chi^2$ kernel and histogram intersection (Hist.-Inte.) kernel. First, the SVM classifier was trained using the SMO algorithm. Then, starting from the obtained discriminative function, the reduction algorithm was tested, for different values of the reduction threshold $\tau$. After each experiment (for each value of $\tau$), the original model was reduced and the number of kept support vectors and the performance of the reduced

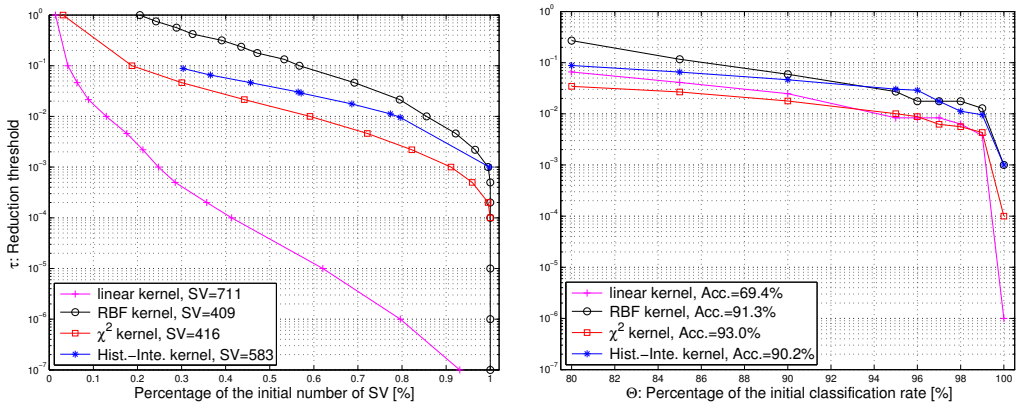



Fig. 8. Percentage of the reduced number of Support Vectors (SV) compared to the initial model (left), and the percentage of the original classification rate that is preserved after the reduction (right), both as a function of different value of $\tau$ for various kernel types. The initial number of Support vectors (SV) and initial classification rate (Acc.) were reported for each kernel.

model were tested on the same test set. If the classification rate dropped below 80% of the initial classification rate, i.e. $\Theta < 80\%$, the process was stopped. Fig. 8 reports the percentage of the reduced number of Support Vectors (SV) compared to the initial model (left), and the percentage of the initial classification rate that is preserved after the reduction (right), as a function of different value of $\tau$. We see that, apart for the linear kernel, the algorithm behaves as expected, obtaining a gentle decrease in performance as the number of stored support vectors is being reduced. It is worth noting that the linear kernel is known for being not a good metric for histogram-like features, as instead all the other three kernels are. This might explain its different behavior.

## 7 Experiments on Adaptation

As a first application of our method, we present experiments on visual place recognition in highly dynamic indoor environments. We consider a realistic scenario, where places change their visual appearance because of varying illumination conditions or human activity. Specifically, we focus on the ability of the recognition algorithm to adapt to these changes over long periods of time. As it is not possible to predict in advance the type of changes that will occur, adaptation must be performed incrementally.

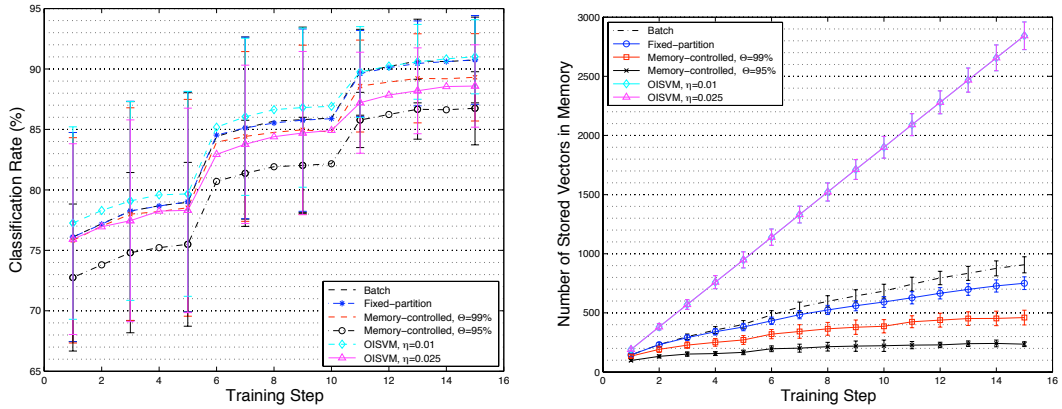
We conducted two series of experiments to evaluate the effectiveness of the memory-controlled incremental SVM for this task. In the first, we considered a case in which the variability observed by the recognition system was *constrained* to changes introduced by long-term human activity under stable illumination conditions. Such experimental procedure allowed us to thoroughly

examine the properties of each of the incremental methods in a more controlled setting. The corresponding experiments are reported in Section 7.1. In the second, we considered a real-world, *unconstrained* scenario where the algorithms had to incrementally gain robustness to variations introduced by changing illumination and short-term human activity, and then, to use their adaptation abilities to handle long-time environment changes. The corresponding experiments are reported in Section 7.2. In both experiments, we compared our approach with the fixed-partition incremental SVM, OISVM and the batch method. This last algorithm is used here purely as a reference, as it is not incremental. We used CRFH global image features. We tested a wide variety of combinations of image descriptors, with several scale levels [37]. On the basis of an evaluation of performance and computational cost, we built the histograms from normalized Gaussian derivative filters applied to the images at two different scales, and we used $\chi^2$ as a kernel for SVM. We also performed experiments using SIFT local features combined with the matching kernel for SVM. Both types of features previously proved effective for the place recognition task [41,40].

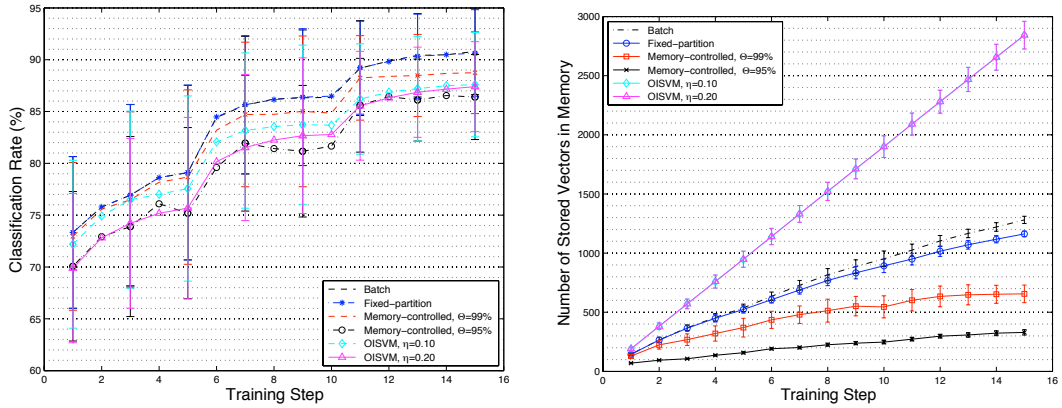
### *7.1 Experiments with Constrained Variability*

In the first series of experiments, we evaluated the properties of the memory-controlled incremental SVM in a simplified scenario. We therefore trained the system on three sequences acquired under similar illumination conditions, with the same robot platform. The fourth sequence was used for testing. Training on each sequence was performed in 5 steps, using one subsequence at a time, resulting in 15 steps in total. We considered 36 different permutations of training and test sequences. Here we report average results obtained on both global and local features by the three incremental algorithms (fixed-partition, OISVM, and memory-controlled) as well as the batch method. We tested the memory-controlled algorithm using two different values of the parameter $\Theta$, i.e. $\Theta = 99\%, 95\%$. This corresponds to the maximum accepted reduction of the recognition rate of 1% and 5% respectively, as explained in Section 4.3. Similarly for OISVM, we used three different values of the parameter $\eta$ that determines how sparse the final solution is going to be (as in [36]).

Fig. 9, left, shows the recognition rates obtained at each incremental step by all methods and for both feature types. Fig. 9, right, reports the number of training samples that had to be stored in the memory at each step of the incremental procedure. First, we see that OISVM achieves very good performance similar to the batch method. However, both methods suffer from the same problem: they require all the training samples to be kept in the memory during the whole learning process. This makes them unsuitable for realistic scenarios, particularly in cases when the algorithm should be used




(a) Classification rate and number of training samples stored for global features.




(b) Classification rate and number of training samples stored for local features.

Fig. 9. Average results obtained for the experiments with constrained variability for three incremental methods and the batch algorithm.

on a robotic platform with intrinsically limited resources. The fixed-partition algorithm achieves identical performance as the batch method, while greatly reducing the number of training samples that need to be stored in the memory at each incremental step. However, despite that all the algorithms show plateaus in the classification rate whenever the model is trained on similar data (coming from consecutive subsequences), the number of support vectors grows roughly linearly with the number of training steps.

We see that for the memory-controlled incremental SVM, both the classification rate and the number of stored support vectors show plateaus every five incremental steps (as opposed to the classification rate only in case of the other methods). The method controls the memory growth much more successfully than the original fixed-partition incremental technique. For instance, when we accept only one percent reduction in classification (i.e. $\Theta = 99\%$), the number of support vectors stored after the 15 steps is 39.6% (CRFH) and 43.7% (SIFT) lower than for the fixed-partition incremental method. For $\Theta = 95\%$, the gain in memory compression is much greater than the overall decrease

in performance. This feature, i.e. the possibility to trade memory for a controlled reduction in performance, can be potentially very useful for systems operating in realistic, open-ended learning scenarios and with limited memory resources. This approach would be even more appealing for systems which can compensate the loss in performance by doing information fusion over time or from multiple sensors. It is worth underlying that the growth in the number of support vectors decreases over time (Fig. 9, bottom). For example, for CRFH and $\Theta = 99\%$, the model trained on the second sequence (step 6 to 10) grows by 115 vectors on average, but trained on the third sequence (step 11 to 15) grows only by 74 vectors. This may indicate that the number of SVs eventually tends to reach a plateau.

In order to gain a better understanding of the methods' behavior, we performed an additional analysis of the results. Fig. 11b shows, for the two approximate incremental techniques, the average amounts of vectors (originating from each of the three training sequences) that remained in the model after the final incremental step (note that, in our case, this analysis would be pointless for OISVM, as it requires storing all the training data). The figure illustrates how the methods weigh instances, learned at different time, when constructing the internal representation. We see that both fixed-partition and memory-controlled algorithms privilege new data, as the SVs from the last training sequence are more represented in the model. This phenomenon is stronger for the memory-controlled algorithm.

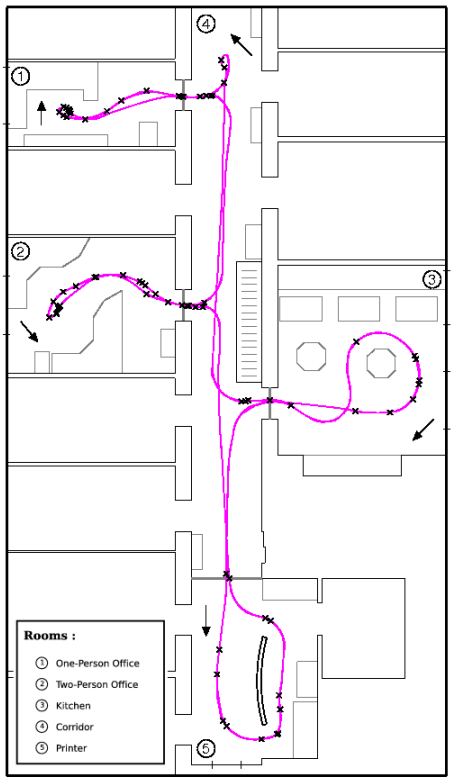
To get a feeling for how the forgetting capability works in case of the memory-controlled method, we plotted the positions where the SVs were acquired, for $\Theta = 99\%$ and the CRFH features. Fig. 10 reports results obtained for a model built after the final incremental step. The positions were marked on three maps presented in Fig. 10a,b,c so that each of the maps shows the SVs originating from only one training sequence. These SVs could be considered as landmarks selected by the visual system for the recognition task. As already shown in Fig. 11b, most of the vectors in the model come from the last training sequence. Moreover, the number of SVs from the previous training steps decreases monotonically, thus the algorithm gradually forgets the old knowledge. It is interesting to observe how the vectors from each sequence are distributed along the path of the robot. On each map, the places crowded with SVs are mainly transition areas between the rooms, regions of high variability, as well as places at which the robot rotated (thus providing a lot of different visual cues without changing position). To illustrate the point, Fig. 11a shows sample images acquired in the corridor, for which the SVs decay quickly, and one of the offices, for which they are being preserved much longer. The results indicate that the forgetting is not performed randomly. On the contrary, the algorithm tends to preserve those training vectors that are most crucial for discriminative classification, and first forgets the most redundant ones.

On the basis of these experimental findings, we can conclude that the memory-controlled incremental SVM is the best method for vision-based robot localization of those considered here. Therefore, in the rest of the paper we will use only this algorithm, with $\Theta = 99\%$.

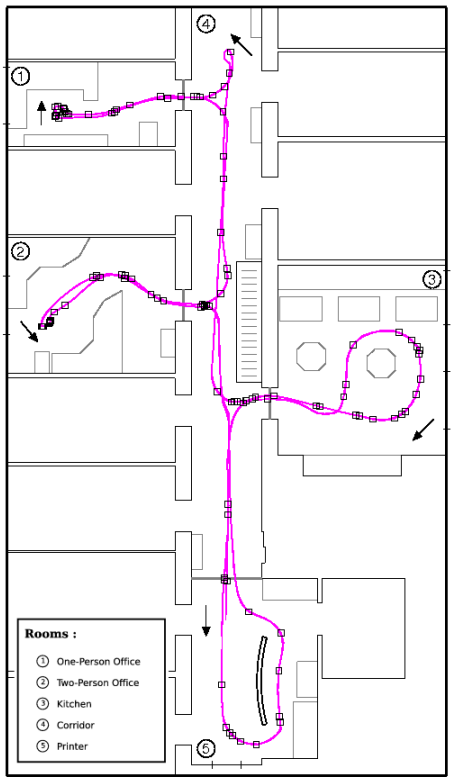
### *7.2 Experiments with Unconstrained Variability*

The next step was to test our incremental method in a real-world scenario. To this purpose, we considered the case where the algorithm needed to incrementally gain robustness to variations introduced by changing illumination and human activities, while at the same time using its adaptation ability to handle long-time changes in the environment. We performed the experiments first on the IDOL2 database. Then, to confirm the behavior on a different set of data, we used the COLD-Freiburg database. We first trained the system on three IDOL2 sequences acquired at roughly similar time but under different illumination conditions. Then, we repeated the same training procedure on sequences acquired 6 months later. In order to increase the number of incremental steps and differentiate the amount of new information introduced by each set of data, each sequence was again divided into five subsequences. In total, for each experiment we performed 30 incremental steps. Since the IDOL2 database consists of pairs of sequences acquired under roughly similar conditions, each training sequence has a corresponding one which could be used for testing. Feature-wise, here we used only the global features (CRFH). Indeed, the experiments presented in the previous section showed that local features achieve an accuracy similar to that of CRFH, but at a much higher computational cost and memory requirement. Also, preliminary experiments show that this behavior is confirmed in this scenario, hence the choice to use here only the global descriptor.

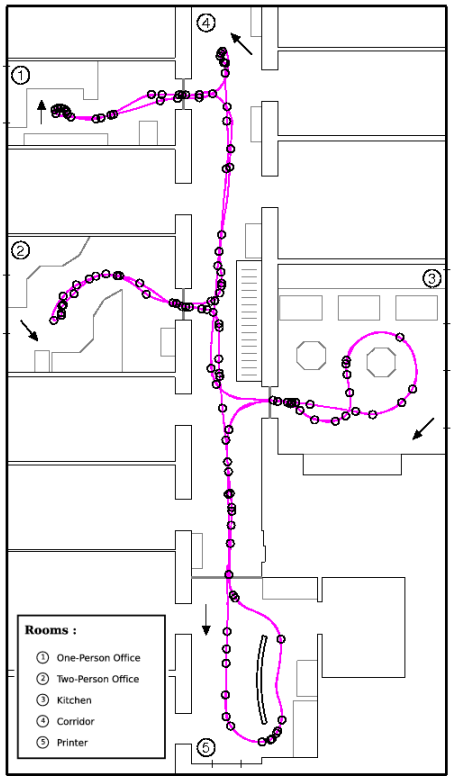
We used a very similar system and experimental procedure for the experiments with the COLD-Freiburg dataset. As in case of IDOL2, we divided each sequence into 5 subsequences and used pairs of sequences acquired under roughly similar conditions for training and testing. In case of both databases, the experiment was repeated 12 times for different orderings of training sequences. Fig. 12 and 13 report the average results together with standard deviations. By observing the classification rates for a classifier trained on the first sequence only, we see that the system achieves best performance on a test set acquired under similar conditions. The classification rate is significantly lower for other test sets. In case of IDOL2, this is especially visible for images acquired 6 months later, even under similar illumination conditions. At the same time, the performance greatly improves when incremental learning is performed on new batches of data. The classification rate decreases for the old test sets; at the same time, the size of the model tends to stabilize.




(a) 78 Support Vectors from 1st seq.

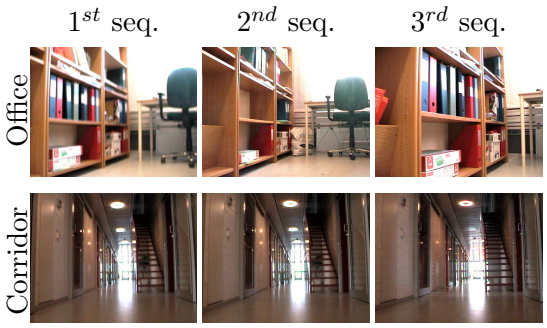



(b) 111 Support Vectors from 2nd seq.

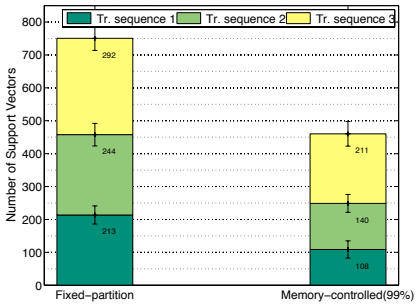



(c) 149 Support Vectors from 3rd seq.

Fig. 10. Maps of the environment with plotted positions of the support vectors stored in the model obtained after the final incremental step for one of the experiments conducted using the memory-controlled technique with $\Theta = 99\%$. The support vectors were divided into three maps (a, b, and c) according to the training sequence they originate from. Additionally, each map shows the path of the robot during acquisition of the sequence (arrows indicate the direction of driving). We observe that the Support Vectors from the old training sequences were gradually eliminated by the algorithm and this effect was stronger in regions with lower variability.

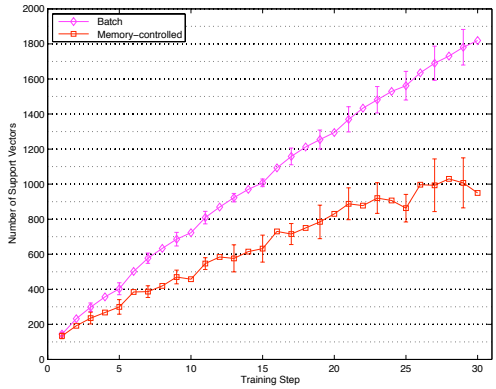



(a) Sample images from the three training sequences.

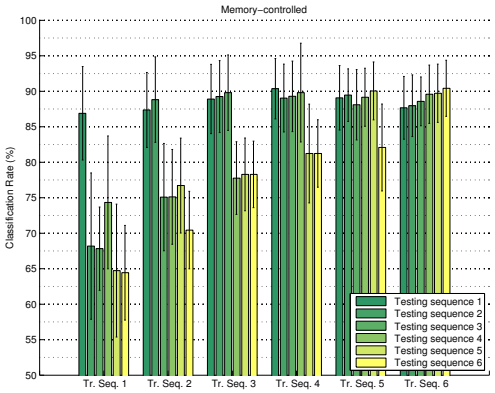



(b) Statistics of Support Vectors stored in the final approximate incremental models.

Fig. 11. Sample images captured in regions of different variability (left). Comparison of the average amounts of training vectors coming from the three sequences that were stored in the final incremental model for the two approximate incremental techniques (right).




(a) Support vectors

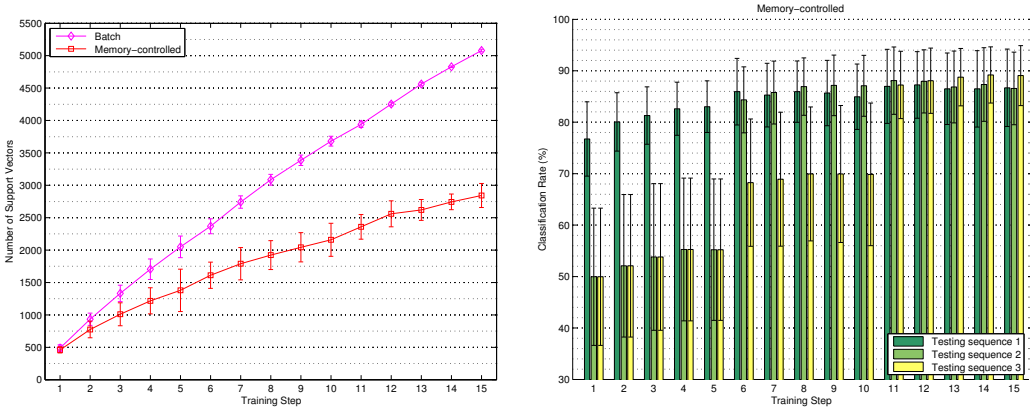



(b) Performance of memory-reduced

Fig. 12. Average results of the IDOL2 experiments in the real-world scenario. (a) compares the amounts of SVs stored in the models at each incremental step for the batch and the memory-controlled method. (b) reports the classification rate measured every fifth step (every time the system completes learning a whole sequence) with all the available test sets. The training and test sets marked with the same indices were acquired under similar conditions.

### *7.3 Discussion*

The presented results provide a clear evidence of the capability of the discriminative methods to perform incremental learning for vision-based place recognition, and their adaptability to variations in the environment. Table 1 summarizes the performance obtained by each method in terms of accuracy, speed, controlled memory growth and forgetting capability. For each algorithm (i.e. for each row), we put a cross corresponding to the property (i.e.




(a) Support vectors (b) Performance of memory-reduced

Fig. 13. Average results of the COLD-Freiburg experiments in the real-world scenario. (a) compares the amounts of SVs stored in the models at each incremental step for the batch and the memory-controlled method. (b) reports the classification rate measured every step with all the available test sets. The consecutive training and testing sequences were acquired under similar conditions.

the column) that the algorithm has shown to possess in our experiments. The fixed-partition method performs as well as batch SVM, but it is unable to control the memory growth and requires much more memory space. We also found that OISVM could get very good accuracy while achieving a low computational complexity during testing. However, none of the two methods has shown to possess an effective forgetting capability: for the fixed-partition method, the old SVs decay slowly, but the decay is neither predictable nor controllable; for OISVM, every training vector must be stored into memory. As opposed to this, the memory-controlled algorithm is able to achieve performances statistically equivalent to those of batch SVM, while at the same time providing a principled and effective way to control the memory growth. Experiments showed that this has induced a forgetting capability which privileges newly acquired data to the expenses of old one and the model growth slows down whenever new data are similar to those already processed. Furthermore, since a lot of training images can be discarded during the incremental process, the training time soon becomes significantly lower than for the batch method. For instance, in case of the second experiments, training the classifier at the last step took 25.5s for the batch algorithm and only 5.6s for the memory-controlled method on a 2.6GHZ Pentium IV machine, and recognition time was twice as fast for the memory-controlled algorithm than for the batch one.

| | Accuracy | Forgetting | Memory | Speed |
|---|---|---|---|---|
| Fixed-partition | x | x | | |
| OISVM | x | | | x |
| Memory-controlled | **x** | **x** | **x** | **x** |

Table 1
Comparing incremental learning techniques for place recognition and robot localization applications.

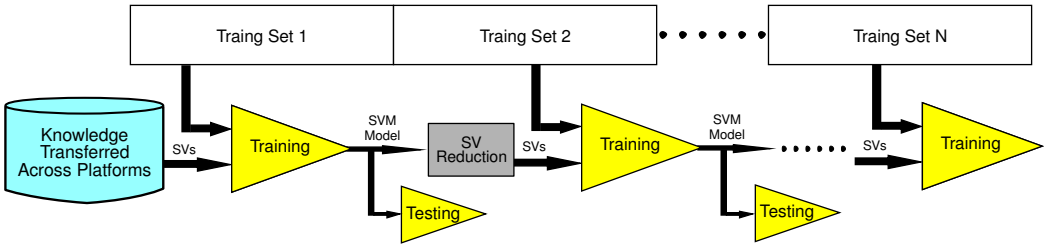



Fig. 14. A diagram illustrating the data flow in the knowledge-transfer system.

## 8 Experiments on Knowledge Transfer

As a second application of our method, we considered the problem of transfer of knowledge between robotic platforms with different characteristics, performing vision-based recognition in the same environment. We used the IDOL2 database and the robots Minnie and Dumbo for these experiments. The main difference between the two platforms lies in the height of the cameras (see Fig. 15). They both use the memory-controlled incremental SVM as a basis for their recognition system, thus they share the same knowledge representation. The aim is to efficiently exploit the knowledge acquired e.g. by one robot so to boost the recognition performance of another robot. We propose to use our method to update the internal representation when new training data are available. Fig. 14 illustrates how our approach can be used for transfer of knowledge. We would like the knowledge transfer scheme to be adaptive, and also to privilege newest data so to avoid accumulation of outdated information. Finally, the solution obtained starting from a transferred model should gradually converge to the one learned from scratch, not only in terms of performance but also of required resources (e.g. memory).

The challenges in the transfer of knowledge will come from:

- *(a) Differences in the parameters of the two platforms*
  The cameras are mounted at two different heights, thus the informative content of the images acquired by the two platforms is different. Because of this, the knowledge acquired by one platform might not be helpful for the other one or, in the worst case, it might constitute an obstacle. Preliminary experiments showed that SIFT is more suitable for the transfer of knowledge

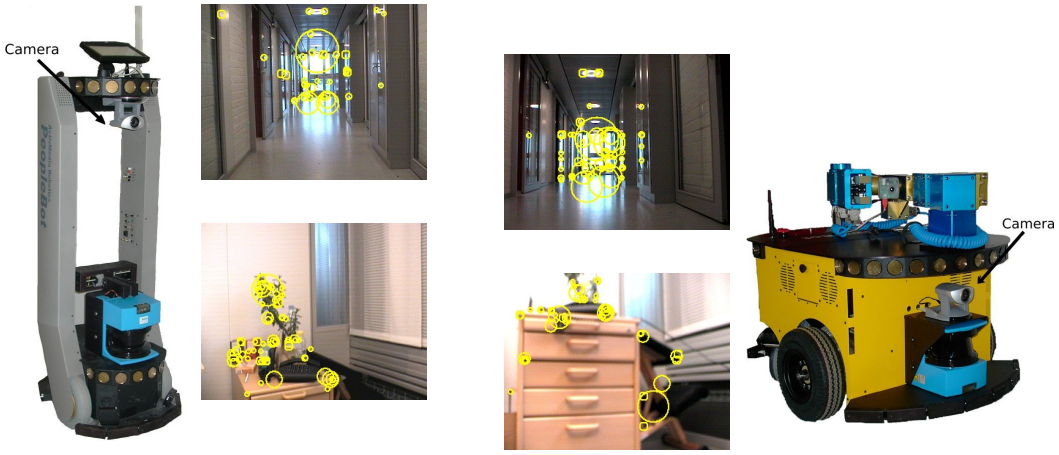



Fig. 15. Knowledge transfer across robot platforms which only partially share visual information.

in our scenario than CRFH. For that reason, CRFH will not be used.

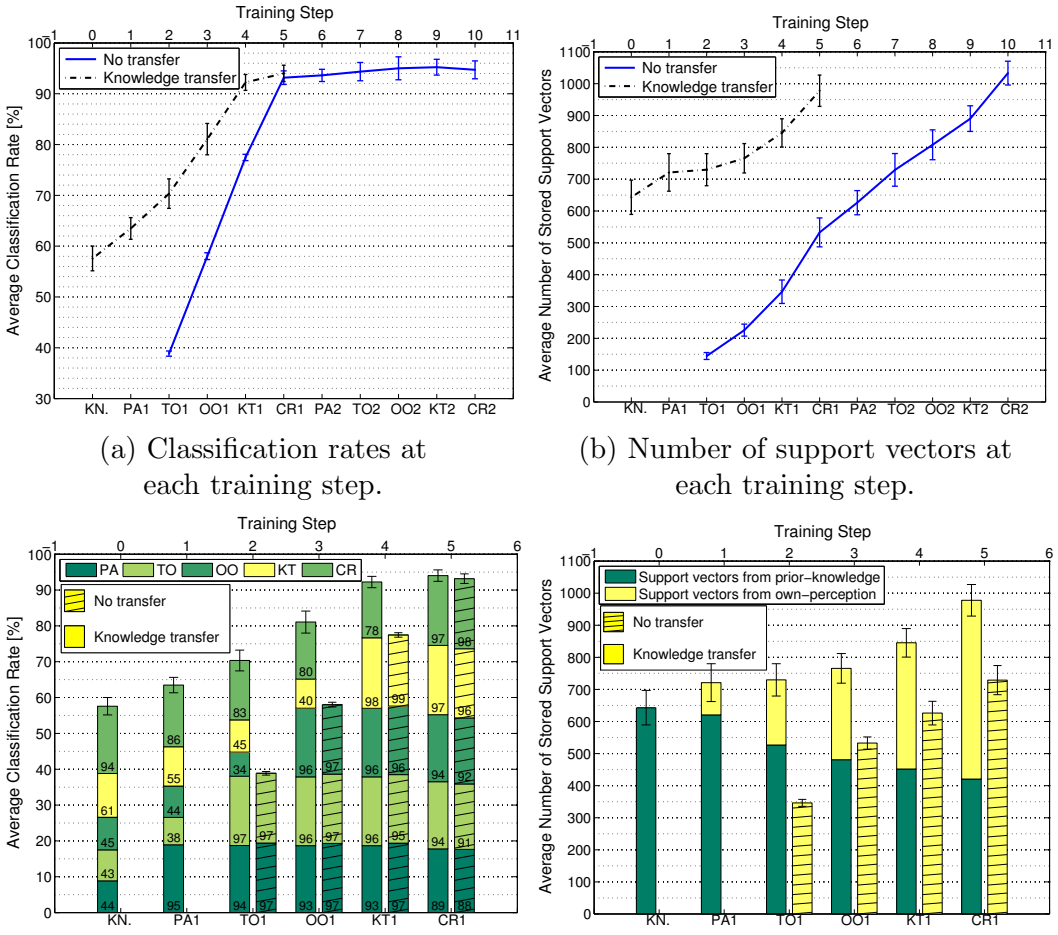
- *(b) Room by room/frames by frames knowledge update*
  It is desirable to update the model transferred across platforms as soon as new data are available. We will investigate the behavior of the algorithm when the update is performed room-by-room, or frames-by-frames. Both scenarios are at risk of unbalanced data with respect to the class being updated.
- *(c) Growing memory requirements*
  Building on top of an already trained classifier might lead to a solution that will be much more demanding in terms of memory usage and computational power than the one learned from scratch. Although our memory-controlled approach is capable of reducing the number of SVs, its reduction process does not take the sources of the information into consideration. In order to favor information coming from the platform currently in use, we imposed to the algorithm to discard only those SVs that were linearly dependent *and* came from the previous platform by adding meta-information on the training examples. This scheme speeds up the turnover of stored SVs, while preferring newest data and at the same time preserving relevant information.

In the IDOL2 database, for each robot and for every illumination condition, we always have two sequences acquired under similar conditions. Here, we always used such pairs of sequences, one as a training set and the other one as a test set. In all the experiments, we benchmarked against a system not using any prior knowledge.

In the first series of experiments, the system was updated incrementally in a room by room (i.e. class by class) scenario. The system was trained incrementally on one sequence; the corresponding sequence, acquired under roughly similar conditions, was used for testing. The prior-knowledge model was built using standard batch SVM from one image sequence, acquired under the same illumination conditions and at close time as the training one, but using a different platform. As there are five classes in total, training was performed in 5 steps (the algorithm learned incrementally one room at the time). In the no-transfer case, the system needed to build the model from scratch, and thus needed to acquire data from at least two classes. In this case, training on each sequence required only 4 steps since in the first step the algorithm learned to distinguish between the first two classes.

Building on top of knowledge acquired from another platform implies a growth in the memory requirements. To evaluate this behavior in relationship to its effects on performance and compare fairly to the system trained without a prior model, we incrementally updated the model without transferred knowledge on another sequence acquired under conditions similar to that of the first training sequence. This experiment makes it possible to evaluate performance and memory growth when both systems are trained on two sequences. The main difference is that in one case both sequences were acquired and processed by the same platform; in the other case, one sequence was acquired and processed by a different platform. We considered different permutations in the rooms order for the updating; for each permutation, we considered 6 different orderings of the sequences used as training, testing, and prior-knowledge sets. Due to space reasons, we report only average results for one permutation, together with standard deviations in Fig. 16.

We can see that, for both approaches, the system gradually adapts to its own perception of the environment. It is clear that the knowledge-transfer system has a great advantage in terms of performance over the no-transfer system at the first steps. For instance, we see that, after the second update (TO1, Fig 16a), the knowledge-transfer system achieves a classification rate of 65.3%, while the no-transfer knowledge obtains only 37%. The advantage in classification rate for the knowledge-transfer system remains considerable for the steps OO1 and KT1. However, it is interesting to note that even when both systems have been updated on a full sequence (CR1, Fig 16a), the knowledge-transfer system still maintains an advantage in performance. Considering the differences between the two platforms, and that the transferred knowledge model was built on a single sequence, this is a remarkable result. It can also be observed from Fig. 16d that the memory-controlled algorithm facilitated the decay of knowledge from the other platform (in the first incremental step,




(a) Classification rates at each training step.

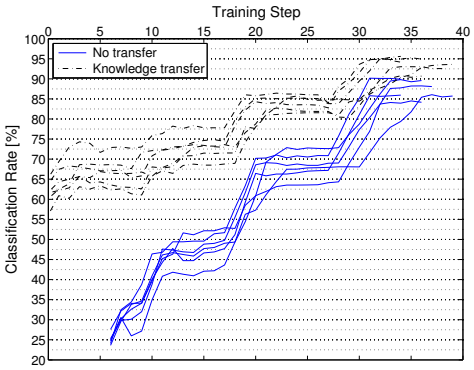
(b) Number of support vectors at each training step.

(c) Comparison of the performance at each training step.

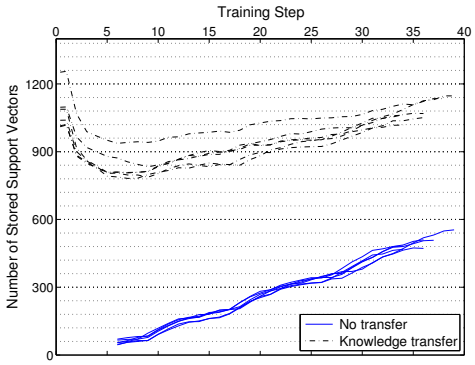
(d) Comparison of the number of support vectors at each step.

Fig. 16. Average results obtained for the system incrementally trained with and without transfer of knowledge in the room by room fashion. Fig. 16a,b compare the final recognition rates and the total number of support vectors for both cases. Fig. 16c,d present a detailed analysis: classification rates obtained for each of the rooms and the amount of support vectors in the final model that originate from the transferred knowledge. In all the plots, the first step "KN." corresponds to the results obtained for the transferred knowledge before any update was performed.

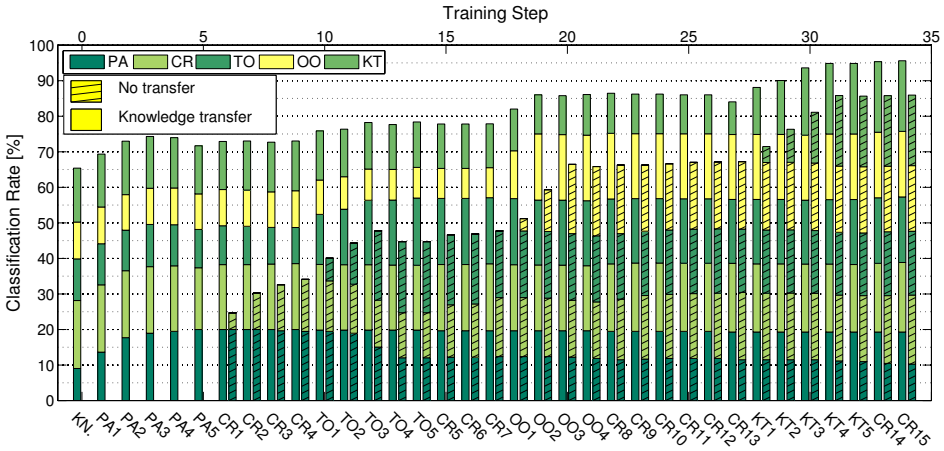
we did not perform the reduction), while the knowledge acquired by its own sensor gradually becomes the main source for the model. As the no-transfer system continued to learn one additional sequence incrementally, its memory growth eventually exceeded the knowledge-transfer case (see Fig 16b). Although the model was built on two sequences acquired by the same platform, the knowledge-transfer system still obtains a comparable performance. We conclude that the transfer of knowledge, in a room by room updating scenario, acts as an effective boosting of performance, without any long-term growth of the memory requirements.



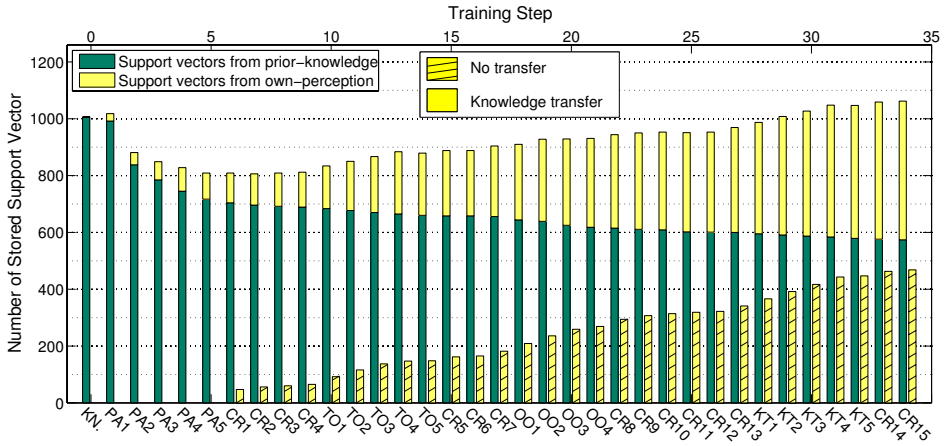

(a) Classification rates at each training step.




(b) Number of support vectors at each training step.




(c) Detailed comparison of the performance of the system with and without knowledge-transfer.




(d) Number of stored support vectors of incremental experiment with and without knowledge-transfer at each step.

Fig. 17. Average results obtained for the system incrementally trained with and without transfer of knowledge in the frames by frames fashion. The labels below each bar indicate the batch of data used for the incremental update. Again, the first step labeled as "KN." corresponds to the results obtained for the transferred knowledge before any update was performed.

The second series of experiments explored the behavior of the system in a frames by frames updating scenario. Here, for each incremental update, we used a certain number of consecutive frames taken from the training image sequence. Again, the system was trained incrementally on one sequence, and a corresponding sequence was used as a test set. We examined the performance of the system for the case when updating was performed using 30 frames per step [1]. Thus, for each experiment, it took more than 30 incremental steps in total to complete a sequence. The prior-knowledge model was built using two complete sequences acquired by the other platform, under the same illumination conditions and very close in time. This provided a better start-up performance than in case of the previous experiments. Again, we benchmarked against the system not using any prior knowledge. In this case, in order to fulfill the requirement of training using at least 2 classes, the first training set consisted of all the images captured in the first room plus the first 30 frames captured in the second room. As a consequence, the full training process required five to six less steps than in case of equivalent experiments using the knowledge-transfer scheme. The experiment was repeated 6 times for different orderings of training sequences. Since the number of training steps varied (due to a different number of images in each sequence), we report all the results separately. Fig. 17a,b report the amount of stored SVs and classification rates at each step, for all the experiments. This shows the general behavior for both approaches. Fig. 17c,d present results for one of the 6 experiments, so to allow a detailed analysis.

By observing the classification rates obtained at each step in both cases, we see that the advantage of the knowledge-transfer scheme is even more visible here than for the room by room updating scenario. This might be due to the fact that some of the training sets used for the no-transfer case are highly unbalanced. We can observe from Fig. 17c that the performance of the system for previously learned rooms can drop considerably when a new batch of frames is loaded; this is not the case for the knowledge-transfer system. The twelfth step, when the system was updated with frames from the two-persons office (TO3, Fig. 17c), is a typical example. Note that this is a general phenomenon present, although less pronounced, also in the room by room updating scenario. Our interpretation is that the model of the prior-knowledge contains information about the overall distribution of the data. This helps to find a balanced solution when dealing with non-separable instances using soft-margin SVM [10]. As a last remark the knowledge from the transferred model is gradually removed over time (see Fig. 17d).

[1] Experiments conducted for 10 and 50 frames per training step gave analogous results, and for space reasons are not reported here.

## 9 Summary and Conclusions

In this paper we presented a novel extension of SVM to incremental learning that achieves the same recognition performance of the standard, batch method while limiting the memory growth over time. This is achieved by discarding, at each incremental step, all the support vectors that are not linearly independent. The information they carry is not lost, as it is retained into the algorithm's decision function in the form of weighting coefficients of the remaining support vectors. We call this method memory-controlled incremental SVM. We applied it to the problem of place recognition for robot topological localization, focusing on two distinct scenarios: adaptation in presence of dynamic changes and transfer of knowledge between two robot platforms engaged in the same task. Experiments show clearly the effectiveness of our approach in terms of accuracy, speed, reduced memory and capability to forget redundant, outdated information.

We plan to extend this work in several ways. First, we want to use the memory-controlled algorithm in multi-modal learning scenarios, for instance using laser-based features combined with visual ones, as done in [41], in an incremental setting. Here we should be able to exploit fully the properties of the method, and aggressively trade memory for accuracy on single modalities, while retaining an high overall performance. Second, we would like to investigate further the knowledge transfer scenario, and incorporate in our framework ways to select the data to be transferred, as proposed in [25]. Future work will concentrate in these directions.

*Acknowledgments*

This work was sponsored by the EU FP7 project CogX (A. Pronobis) and IST-027787 DIRAC (B. Caputo, L. Jie), and the Swedish Research Council contract 2005-3600-Complex (A. Pronobis). The support is gratefully acknowledged.

## References

[1] COGNIRON: The cognitive robot companion. Website: `http://www.cogniron.org`.

[2] CoSy: Cognitive Systems for Cognitive Assistants. Website: `http://www.cognitivesystems.org/`.

[3] RobotCub. Website: `http://www.robotcub.org/`.

[4] M. Artač, M. Jogan, and A. Leonardis. Mobile robot localization using an incremental eigenspace model. In *Proc. ICRA'02*.

[5] S. Belongie, C. Fowlkes, Chun F., and J. Malik. Spectral partitioning with indefinite kernels using the nystrom extension. In *Proc. of ECCV'02*.

[6] Emma Brunskill, Thomas Kollar, and Nicholas Roy. Topological mapping using spectral clustering and classification. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, San Diego, October 2007.

[7] B. Caputo, E. Hayman, and P. Mallikarjuna. Class-specific material categorisation. In *Proc. ICCV'05*.

[8] G. Cauwenberghs and T. Poggio. Incremental and decremental support vector machine learning. In *Proc. NIPS'00*.

[9] Chih Chung Chang and Chih Jen Lin. *LIBSVM: A Library for Support Vector Machines*, 2001. Available at `http://www.csie.ntu.edu.tw/~cjlin/libsvm`.

[10] N. Cristianini and J. S. Taylor. *An introduction to support vector machines and other kernel-based learning methods*. Cambridge University Press, 2000.

[11] A. Leonardis D. Skočaj. Weighted and robust incremental method for subspace learning. In *Proc. ICCV'03*.

[12] M. Dissanayake, P. Newman, S. Clark, H.F. Durrant-Whyte, and M. Csorba. A solution to the simultaneous localization and map building (SLAM) problem. *IEEE Transactions on Robotics and Automation*, 17(3):229–241, 2001.

[13] C. Domeniconi and D. Gunopulos. Incremental support vector machine construction. In *Proc. ICDM'01*.

[14] Gyuri Dorkó and Cordelia Schmid. Object class recognition using discriminative local features. 2005.

[15] Tom Downs, Kevin E. Gates, and Annette Masters. Exact simplification of support vector solutions. *J. Mach. Learn. Res.*, 2, 2002.

[16] J. Folkesson, P. Jensfelt, and H. Christensen. Vision SLAM in the measurement subspace. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA'05)*, pages 30–35, Barcelona, Spain, 2005.

[17] M. Fritz, B. B. Leibe, B. Caputo, and B. Schiele. Integrating representative and discriminant models for object category detection. In *Proc. ICCV'05*.

[18] G. H. Golub and C. F. Van Loan. *Matrix Computations*. Johns Hopkins University Press, 3rd edition, 1996.

[19] M. Jogan and A. Leonardis. Robust localization using an omnidirectional appearance-based subspace model of environment. *Robotics and Autonomous Systems*, 45(1):51–72, October 2003.

[20] T. Darrel K. Grauman. The pyramid match kernel: discriminative classification with sets of image features. In *Proc. ICCV'05*.

[21] George Konidaris and Andrew G. Barto. Autonomous shaping: knowledge transfer in reinforcement learning. In *Proc. of ICML'06*.

[22] D. Kortenkamp and T. Weymouth. Topological mapping for mobile robots using a combination of sonar and vision sensing. In *Proceedings of the 12th National Conference on Artificial Intelligence*, Seattle, Washington, USA, 1994.

[23] Geert-Jan M. Kruijff, Hendrik Zender, Patric Jensfelt, and Henrik I. Christensen. Situated dialogue and spatial organization: What, where... and why? *International Journal of Advanced Robotic Systems (ARS), Special Issue on Human-Robot Interaction*, 4(1):125–138, March 2007.

[24] B. Kuipers and P. Beeson. Bootstrap learning for place recognition. In *Proceedings of the 18th National Conference on Artificial Intelligence (AAAI'02)*, 2002.

[25] A. Lazarich, M. Restelli, and A. Bonarini. Transfer of samples in batch reinforcement learning. In *Proc. ICML08*.

[26] O. Linde and T. Lindeberg. Object recognition using composed receptive field histograms of higher dimensionality. In *Proceedings of the 17th International Conference on Pattern Recognition (ICPR'04)*, Cambridge, UK, 2004.

[27] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *Int. J. Computer Vision*, 60(2):91–110, 2004.

[28] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004.

[29] J. Luo, A. Pronobis, B. Caputo, and P. Jensfelt. The IDOL2 database. Technical Report 304, CVAP, KTH, 2006. Available at `http://cogvis.nada.kth.se/IDOL2/`.

[30] Jr. Malak, R.J. and P. K. Khosla. A framework for the adaptive transfer of robot skill knowledge using reinforcement learning agents. In *Proc. of ICRA'01*.

[31] O. Martínez Mozos, R. Triebel, P. Jensfelt, A. Rottmann, and W. Burgard. Supervised semantic labeling of places using information extracted from sensor data. *Robotics and Autonomous Systems*, 55(5), 2007.

[32] K. Mikolajczyk and C. Schmid. Indexing based on scale invariant interest points. In *Proceedings of the 8th IEEE International Conference on Computer Vision (ICCV'01)*, Vancouver, Canada, 2001.

[33] Tom Mitchell. The discipline of machine learning. Technical Report CMU-ML-06-108, CMU, 2006.

[34] A. C. Murillo, J. Kosecka, J. J. Guerrero, and C. Sagues. Visual door detection integrating appearance and shape cues. *Robotics and Autonomous Systems*, 56(6):pp. 512–521, June 2008.

[35] Illah Nourbakhsh, Rob Powers, and Stan Birchfield. Dervish: An office navigation robot. *AI Magazine*, 16(2):53–60, 1995.

[36] F. Orabona, C. Castellini, B. Caputo, J. Luo, and G. Sandini. Indoor place recognition using online independent support vector machines. In *18th British Machine Vision Conference (BMVC07)*, Warwick, UK, September 2007.

[37] A. Pronobis. Indoor place recognition using support vector machines. Master's thesis, NADA/CVAP, Kungliga Tekniska Hoegskolan, Stockholm, Sweden, December 2005.

[38] A. Pronobis and B. Caputo. Confidence-based cue integration for visual place recognition. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'07)*, San Diego, CA, USA, October 2007.

[39] A. Pronobis and B. Caputo. COLD: COsy Localization Database. *The International Journal of Robotics Research (IJRR)*, 28(5), May 2009.

[40] A. Pronobis, B. Caputo, P. Jensfelt, and H. I. Christensen. A discriminative approach to robust visual place recognition. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'06)*, Beijing, China, October 2006.

[41] A. Pronobis, O. Martínez Mozos, and B. Caputo. SVM-based discriminative accumulation scheme for place recognition. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA'08)*, Pasadena, CA, USA, May 2008.

[42] S. Se, D. G. Lowe, and J. Little. Vision-based mobile robot localization and mapping using scale-invariant features. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA'01)*, 2001, Seoul, Korea.

[43] Christian Siagian and Laurent Itti. Biologically-inspired robotics vision monte-carlo localization in the outdoor environment. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'07)*, San Diego, CA, USA, October 2007.

[44] Christian Siagian and Laurent Itti. Biologically-inspired robotics vision monte-carlo localization in the outdoor environment. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'07)*, San Diego, CA, USA, October 2007.

[45] N. A. Syed, H. Liu, and K. K. Sung. Incremental learning with support vector machines. In *Proc. IJCAI'99*.

[46] A. Tapus and R. Siegwart. Incremental robot mapping with fingerprints of places. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'05)*, Edmonton, Alberta, Canada, August 2005.

[47] Sebastian Thrun. Learning metric-topological maps for indoor mobile robot navigation. *Artificial Intelligence*, 1999(1), 1998.

[48] Sebastian Thrun and Tom Mitchell. Lifelong robot learning. *Robotics and Autonomous Systems 15*, 1995.

[49] A. Torralba, K. P. Murphy, W. T. Freeman, and M. A. Rubin. Context-based vision system for place and object recognition. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV'03)*, 2003, Nice, France.

[50] I. Ulrich and I. Nourbakhsh. Appearance-based place recognition for topological localization. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA'00)*, San Francisco, CA, USA, 2000.

[51] Christoffer Valgren and Achim J. Lilienthal. SIFT, SURF and seasons: Long-term outdoor localization using local features. In *Proceedings of the European Conference on Mobile Robots (ECMR'07)*, 2007.

[52] V. Vapnik. *Statistical learning theory.* Wiley and Son, 1998.

[53] C. Wallraven, B. Caputo, and A. Graf. Recognition with local features: the kernel recipe. In *Proc. of ICCV'03.*

[54] J. Wolf, W. Burgard, and H. Burkhardt. Robust vision-based localization by combining an image retrieval system with monte carlo localization. *IEEE Transactions on Robotics*, 21(2):208–216, 2005.

# Mechanical support as a spatial abstraction for mobile robots

Kristoffer Sjöö, Alper Aydemir, Thomas Mörwald, Kai Zhou and Patric Jensfelt

***Abstract*— Motivated by functional interpretations of spatial language terms, and the need for cognitively plausible and practical abstractions for mobile service robots, we present a spatial representation based on the physical support of one object by another inspired by the preposition "on". A perceptual model for evaluating this relation is suggested, and experiments – simulated as well as using a real robot – are presented. We indicate how this model can be used for important tasks such as communication of spatial knowledge, abstract reasoning and learning, exemplifying this in the context of direct and indirect visual search. We also demonstrate the model experimentally, showing that it produces intuitively feasible results from visual scene analysis as well as synthetic distributions that can be put to a number of uses.**

## I. Introduction

The field of service robotics is, at its core, directed toward the creation of systems that are as versatile, adaptive and powerful in everyday environments as human beings are. Only when this becomes true will we be able to depend on robots in the same way as on people around us.

The human machine is superbly adapted to this kind of environment; not just physically (such as having legs to negotiate stairs and thresholds, and arms for opening doors and using appliances), but mentally as well. Human cognition, language, and civilisation have all evolved, and are evolving, in inextricable conjunction with each other. Any cultural or linguistic concept, whether it is the function of a piece of furniture or the meaning of a word, needs the support of cognitive mechanisms; individuals are driven to acquire such mechanisms by reinforcement pressures from their surroundings [8] – while at the same time, the minds of individuals, embodied in the real world, shape and bring forth that same cultural or linguistic concept in turn.

This all suggests the following:

1) Adopting human-like cognitive patterns will help robots approach human-like performance in the context of homes, offices or other environments that are the products of human inclinations, activities and thought.
2) Linguistic concepts can provide insights into cognition that can help understand the nature of those cognitive patterns.

These are the principles on which this work is based. Our research addresses *spatial* concepts specifically. Spatial concepts are of great importance to robotic agents, especially mobile ones:

- They are a necessary part of linguistic interaction with human beings, both when interpreting utterances with a spatial content and when formulating such utterances.
- They allow knowledge transfer between systems, whether different robots, or databases such as the Open Mind Indoor Common Sense database (OMICS) [2] (which contains "commonsense" information about indoor environments provided by humans, such as where objects may be found), as long as those concepts are shared.
- They provide qualitative abstractions that facilitate learning and reasoning.
- They can be used to guide top-down processes such as e.g. visual object search.

Drawing inspiration from results in psycholinguistics, in this paper we examine the functional spatial relation of mechanical support, which in English corresponds to the preposition "on". We contribute a novel and general perceptual measure that allows a robot to analyze a scene in terms of this relation in practice. We implement this perceptual model showing it to produce results in accord with human intuitions of "on"; we also perform simulated sampling experiments to show how it can be used in a top-down fashion to generate a conditional probability distribution over object poses given that the relation is known or assumed to hold.

Other work has examined ways to quantify spatial relations. [11], inspired by findings on spatial information encoding in the hippocampus, suggests a number of geometrical factors, e.g. coordinate inequalities, that play a part in defining relations such as "below", "near" or "behind", but does not attempt to provide exact formulas.

In [13], the *Attention Vector Sum* is proposed as a practical numerical measure of how acceptable a particular spatial relation is for describing a scene, and this model is compared to actual human responses. The scenes used in this work are 2-dimensional and the trajector (mobile object) is treated as a single point.

[10] presents a system where a user can sketch images of basic figures, and which learns to distinguish between examples of "in", "on", "above", "below" and "left". However, the domain used in the work is strictly 2-dimensional.

Topological relations specifically are surveyed in [3]. *Region connection calculus* and its variants provide a language for expressing qualitative relationships between regions, such as containment, tangential contact etc. Relations are of an all-or nothing nature; and they represent objective, geometrical as opposed to perceptual or functional attributes.

The aforementioned work, because of its emphasis on pure geometry – typically in 2 dimensions – is not directly

K. Sjöö, A. Aydemir and P. Jensfelt are with the Centre for Autonomous Systems at the Royal Institute of Technology (KTH), Stockholm, Sweden. T. Mörwald and K. Zhou are with the Automation and Control Institute, Vienna University. This work was supported by the SSF through its Centre for Autonomous Systems (CAS), and by the EU FP7 project CogX and the Swedish Research Council, contract 621-2006-4520 (K. Sjöö)

suited for applications in a practical mobile robotic scenario. This paper, in contrast, takes a novel, functional approach by basing a relation on a single fundamental, objective mechanical property. Another contribution lies in treating all the objects as entire bodies rather than simplifying them into points, a simplification which ignores the importance of physical contact in the "on" relation. We also show how the method can be used to generate probability distributions, such as might be used for visual search.

This paper is organized in the following way: Section II introduces the spatial relation we are examining and the suggested perceptual model for the relation; Section III presents the implementation of the model that we have carried out and the experiments performed – on real image data as well as simulated. Section IV discusses the results and directions for future research, followed by conclusions in Section V.

## II. The On relation

Spatial predicates in language come in different categories. *Projective* spatial relations constrain the trajector's[1] location within an essentially *directed* region relative to the landmark. Examples in English include "to the left of", "behind" and "past". *Topological* relations, in contrast, locate the trajector in some manner that is independent of direction. Typical examples are "on", "at" and "inside". Topological relations seem to be among the first to be learned in humans [12]. In this work, we are concerned with "on", an important English word implying an equally important underlying spatial concept.

Research suggests that verbal descriptions of space do not, in general, correspond one-to-one to cognitive representations [9]. Instead, it seems conceptualization forms around kernels of *functional* criteria, such as "physical attachment", "superposition" (an object being located in the space vertically above another) or "containment" (an object being enclosed by another). As has been noted by e.g. Talmy [15] and Herskovits [6], English' "on" carries a central meaning also represented in many other languages: that of *support* against gravity; i.e., a trajector is "on" a landmark if it would, were the landmark to be removed, begin to fall or move under the influence of gravity. This sense of "on" is an *idealized cognitive model* or ICM [7], around which other, less central and more idiomatic senses of "on" form in a way specific to each language.

### A. The importance of support in robotics

We observe that the notion of support is highly related to the functional aspects of space as designed, constructed and lived in by human beings. Such space is full of entities specifically made to support others, both statically – such as tables, shelves, counters, chairs, hooks and desks – and dynamically – such as trays, trolleys, and dishes. This functional aspect is emphasized by Coventry and Garrod [4]:

> Describing where an object is located goes beyond the description of a geometric position of objects as a snapshot in time. Understanding spatial language is also about the *purpose* that location serves for the users of that language.

As for "on", it is the 14th most common English word [1] which indicates the importance that humans attach to support in representing the spatial location of an object[2].

Apart from the evidence given by its prominent role in language (and thus in the minds of people), support is an intuitively useful abstraction in the following way: If a support is moved, then supported objects will tend to move with it, maintaining the relation (Coventry and Garrod refer to this as "Location control" [4]), and it makes the relation inherently hierarchical, which is a useful property in spatial organization.

Secondly, the fact that artifacts in the environment are explicitly designed to provide support surfaces for objects means that often, when an object is "on" another, it *belongs* there functionally to some degree and is thus likely to be replaced on the same surface even after a human picks up, manipulates, or moves it – even though the exact position may have changed. For example, a desk may be shifted or moved, or worked at by its owner, and its set of supported objects yet be unchanged.

It thus is of interest to robotics to use a spatial representation that encodes this functional relationship between objects. Although this work is inspired by linguistic clues, giving a robot additional linguistic capabilities is only an incidental outcome. It is also necessary to point out that the word "on" spans far more meanings than the core physical support relation: it may entail indirect rather than direct support, adhesive or suspended support, as well as metaphorical uses. Here, we are not attempting to cover that complexity.

### B. A perceptual model

The "support" relation proposed above constitutes an idealized model, but is as such not possible to evaluate directly from perceptual data. Neither robots nor humans can ascertain degree of mechanical support merely by visually regarding a scene, and so it becomes necessary to introduce a perceptual model to estimate the ideal relation.

Humans use context, experience with specific objects and generalizations, as well as schemata to decide whether an object is "on" another. For robots, we model this with a simplified 3-dimensional geometric predicate, termed On, such that $\text{On}(A, B)$ corresponds to "A is supported by B". The relation is graded and can attain values in the range $[0, 1]$.

The following are our criteria and their justification. $O$ denotes the trajector object, and $S$ the support object or landmark. The criteria are illustrated in Figure 1.

[1]The trajector is the entity whose location (and/or motion) is being denoted explicitly, in relation to the landmark. Thus, in the sentence "$A$ is above $B$", $A$ is the trajector and $B$ the landmark.

[2]Though many usages of "on" in English are not about support directly, or even about literal space, the fact that "on" is the word used still underscores the cognitive centrality of its core meaning.

1) *Separation between objects*, $d$. $d$ can be positive or negative, negative values meaning that objects seem to be interpenetrating.
   In order for an object to mechanically support another, they must be in contact. Due to imperfect visual input and other errors, however, contact may be difficult to ascertain precisely. Hence, the apparent separation is used as a penalty.
2) *Horizontal distance between COM and contact*, $l$. It is well known that a body $O$ is statically stable if its center of mass (COM) is above its area of contact with another object $S$; the latter object can then take up the full weight of the former. Conversely, the greater the horizontal distance between the COM and the contact, the less of the weight $S$ can account for, as the torque gravity imposes on $O$ increases, and this torque must be countered by contact with some other object.
   Thus we impose a penalty on $\textsc{On}(O, S)$ that increases with the horizontal distance from the contact to the COM of $O$. The contact is taken to be that portion of $S$'s surface that is within a threshold, $\delta$, of $O$, in order to deal with the uncertainties described above. If $d > \delta$, the point on $S$ closest to $O$ is used instead; otherwise, $l$ is the positive distance to the outer edge of the contact area if outside it, and the negative distance if inside.
3) *Inclination of normal force*, $\theta$ – the angle between the normal of the contact between $O$ and $S$ on the one hand, and the vertical on the other. The reason for including this is that *mutatis mutandis*, the normal force decreases as the cosine of $\theta$, meaning the weight of $O$ must be either supported by another object or by friction (or adhesion).

All these values can be computed from visual perception in principle. Unless otherwise known in advance, the position of the COM is taken as the geometrical centroid of the object (since density cannot be determined by vision).

In order to allow a measurable value to be computed, the agreement with each of the three above criteria is represented as a continuous function, with a maximum at the point of best agreement with the criterion. This provides robustness against error. Criterion 1 is represented by an exponential *distance factor*:

$$\textsc{On}_{distance}(O, S) \triangleq \exp\left(-\frac{d}{d_0(d)} \ln 2\right) \tag{1}$$

where $d_0$ is the falloff distance at which On drops by half.

$$d_0 = \begin{cases} -d_0^-, & d < 0 \\ d_0^+, & d >= 0 \end{cases}$$

The constants $d_0^-$ and $d_0^+$ are both greater than 0 and can have different values (representing the penetrating and nonpenetrating cases, respectively).

Criteria 2 and 3 make up the sigmoid-shaped *contact factor*:

$$\textsc{On}_{contact}(O, S) \triangleq \cos\theta \cdot \frac{1 + \exp(-(1 - b))}{1 + \exp\left(-\left(\frac{-l}{l_{\max}} - b\right)\right)} \tag{2}$$

Here, $l_{\max}$ is the maximum possible distance an internal point can have within the contact area, and $b$ is an offset parameter.

The values are combined by choosing whichever factor is smaller, indicating the greater violation of the conditions for support:

$$\textsc{On}(O, S) \triangleq \min(\textsc{On}_{contact}, \textsc{On}_{distance}) \tag{3}$$

Note that the resultant value of On, although in the range $[0, 1]$, is not a probability. Rather, it represents the degree of resemblance of the visual scene to the prototypical On case. It can be thresholded to produce a true/false judgement, which may in turn be utilised in a qualitative reasoning framework, or for learning – such as learning relationships between object types in an environment. Alternatively, the On measure could be compared with similar measures for other relations or other objects, to determine which linguistic description of the scene is the most apt. It can also be used to weight samples to produce a distribution over poses of $O$, as discussed below.

### C. Probability modelling

The conceptualization above does not explicitly make use of any probabilities. However, it is obvious that the fact of an object being On another is not sufficient to recover the exact pose of the trajector. A probability distribution over poses can be produced in the following way:

Given the pose and geometry of the landmark $S$, and the geometry (but not the pose) of the trajector $O$, each possible pose $\pi$ for the trajector yields a value of $\textsc{On}(O_\pi, S)$ for that pose.

It is now possible to introduce probabilities in the following way. Introduce a true/false event $On(O, S)$ signifying that $\textsc{On}(O, S) > t$ where $t$ is a threshold. Then,

$$p(\pi|On(O_\pi, S)) = \frac{p(On(O_\pi,S)|\pi)p(\pi)}{p(On(O_\pi,S))} = \\ = \frac{[\textsc{On}(O_\pi,S)>t]p(\pi)}{p(On(O_\pi,S))} \tag{4}$$

Here [] denotes the Iverson bracket:

$$[X] = \begin{cases} 1, & \text{if } X \text{ is } \textsc{True} \\ 0, & \text{otherwise} \end{cases}$$

In other words, the probability is simply proportional to the prior for the pose $\pi$ whenever $\textsc{On}(O_\pi, S) > t$, and 0 elsewhere. Though it may be hard to express this distribution analytically, by drawing samples randomly from $p(\pi)$, discarding those failing to reach the threshold, and normalising over the remainder, an arbitrarily good approximation can be found.

### D. Example: Visual object search

One use for the above probabilistic formulation is the task of locating an object by searching for it visually [16], [18], [19]. Visual object search is typically posed as the problem of selecting a series of views $\{V_i\}$, such that the cost of acquiring and processing those views is minimized while detecting the sought object at some set probability.

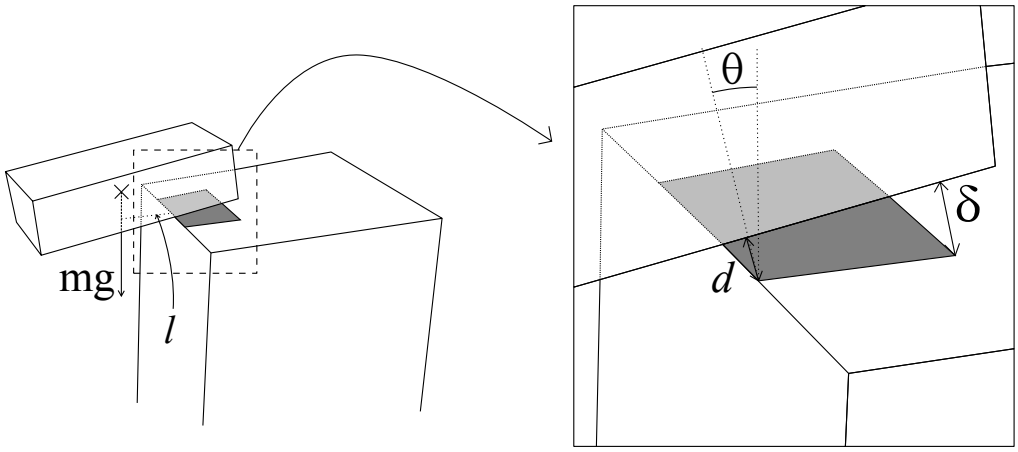



Fig. 1. Key features used in computation of ON: Separation $d$, COM offset $l$, contact angle $\theta$ and contact threshold $\delta$. The gray area represents the contact.

Assume that some algorithm exists that produces a sequence of views, given a probability distribution for the sought object $p(\pi_O = x) = f_O(x)$, the views incurring the total cost $C_O\{f_O\}$. The cost may depend on the actual object, due to size, saliency et cetera.

In this context, the ON relation can be highly useful. In many scenarios, the exact position of an object $O$ may be uncertain or unknown, even while it is known or presumable that it is ON some other object $S$. This information can have several sources: $O$ may have been seen ON $S$ at an earlier time, and location control implies the relation will still hold even if $S$ has moved. The connection may also be statistical in nature, learned through experience from many analysed scenes ("this type of object is usually located ON that type") or from a commonsense knowledge database. The information may also come from symbolic reasoning or linguistical utterances.

Using an object's location to help search for another is known as *indirect search*. Indirect search was first investigated in 1976 by Garvey [5]; there, a system looking for a phone in a room is first tasked with finding the table that the phone is resting on. Wixson [17] re-visited the idea of indirect search in the context of mobile robotics; however, previous work on exploiting spatial relations to guide the visual search process on mobile robots is non-existant.

If it is known a-priori that $On(O, S)$, and the location of $S$ is known, then the above distribution may be used as a prior probability input to a view-selection algorithm, at cost $C_O\{f_{O|S}\}$.

If $On(O, S)$ is known to hold but $S$' location is not known, there are two choices: Indirect search can be used, i.e. locating $S$ first and then locating $O$ given the position of $S$. The cost of this will be[3]:

$$C_S\{f_S\} + C_O\{f_{O|S}\}$$

[3]Although the second term cannot be known exactly without knowing $S$' orientation, one can compute an average over orientations or use a typical orientation; either way, the cost will not vary much for most objects.

Alternatively, one may use a distribution over $O$'s location obtained through chain inference:

$$C_O\{f_O\} = C_O\left\{\int_S f_{O|S} f_S\right\}$$

Either approach can be evaluated using the sampling method suggested above. By comparing the costs, the most beneficial option can be selected depending on the situation.

## III. EXPERIMENTS

To test the feasibility of the concepts described in the preceding section, we have implemented them on a robotic system and tested it in a real-world setting.

We also present a series of simulations that illustrate the potential of the approach using random sampling to synthesize a distribution over positions in space.

### A. Experimental setup

The robot used in our experiments is a Pioneer III wheeled robot, equipped with a stereo camera mounted on a pan-tilt unit at 1.4 m above the ground.

Three different box-shaped objects were used for the tests: A, B and C, as seen in Figs. 2–5. Objects were detected and an initial pose estimated using SIFT features, and the pose refined and tracked using particle filtering based on edge information acquired from the known geometric model of each object [14]. Furthermore, horizontal plane patches were extracted from stereo depth information and assembled into planar objects (table surfaces).

The resulting object poses, along with their known geometries, were then processed by the ON computation described in Section II, using the parameter settings $\delta$ =3 cm, $d_0^+$ =2 cm, $d_0^-$ =1.4 cm and $b = 0.5$. The center-of-mass of each object was taken to be its geometrical center.

### B. Results

Figure 2 shows a simple case (The wireframe contours show the estimated object poses output by the tracking

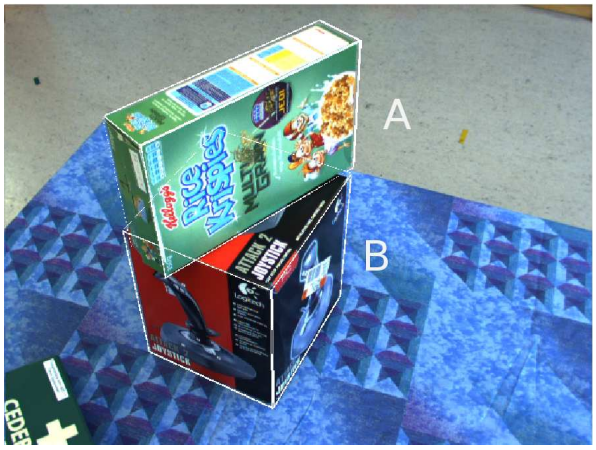



Fig. 2. Typical case: $A$ On $B$, $B$ On table


Fig. 3. Ambiguous case: B partly On A, B partly On table


Fig. 4. Ambiguous case: C partly On each of A and B

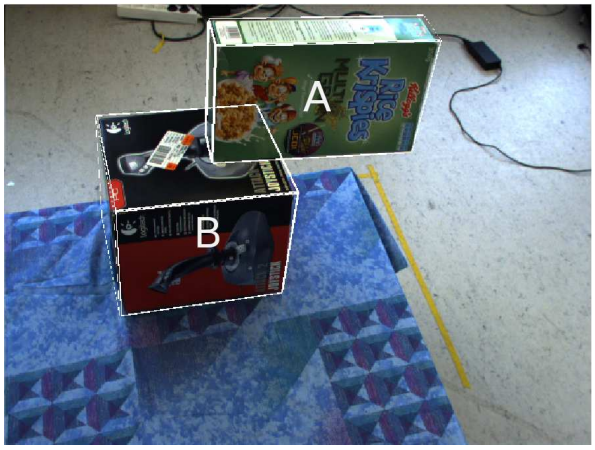



Fig. 5. An anomalous case

algorithm). The values for the support function in this scene are:

| | On$(A, x)$ | On$(B, x)$ |
|---|---|---|
| $x = A$ | — | 0% |
| $x = B$ | 93% | — |
| $x =$Table | 2% | 92% |

The support relation is unambiguous in this case: A is supported by B, and B by the table. In Figure 3, the situation is more ambiguous, with B resting partly on the table, and also leaning on A. The ambiguity is reflected in the On measures:

| | On$(A, x)$ | On$(B, x)$ |
|---|---|---|
| $x = A$ | — | 25% |
| $x = B$ | 0% | — |
| $x =$Table | 74% | 47% |

Figure 4 shows another double support example; the object is held up approximately equally by the two objects, which is reflected in the computed function:

| | On$(A, x)$ | On$(B, x)$ | On$(C, x)$ |
|---|---|---|---|
| $x = A$ | — | 1% | 28% |
| $x = B$ | 0% | — | 30% |
| $x = C$ | 0% | 0% | — |
| $x =$Table | 91% | 93% | 3% |

Finally, Figure 5 depicts a situation that is seemingly physically implausible.

| | On$(A, x)$ | On$(B, x)$ |
|---|---|---|
| $x = A$ | — | 0% |
| $x = B$ | 22% | — |
| $x =$Table | 4% | 84% |

The On measure here is low, and even though there is no other object with which to compare it, the low value means the configuration is far from prototypical and not one that would be expected by the robot, given only the information that "$A$ is on $B$". The problem here is that the COM has been modified with an extra weight to not be at the geometrical center of $A$, but the robot doesn't know this, and as stated earlier it cannot be gleaned from vision alone.

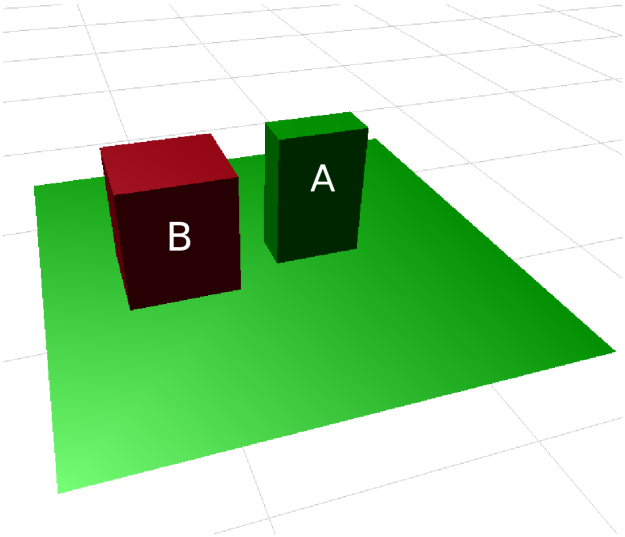



Fig. 6. Objects used in simulation experiments

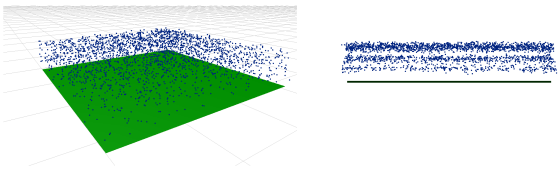


Fig. 7. Position of $A$, given "$A$ On Table"

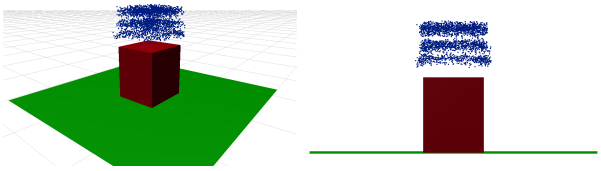


Fig. 8. Position of $A$, given "$A$ On $B$"

In summary, we have verified that our approach works in an implemented real-world system, all the way from sensors to spatial abstraction, producing outputs that are intuitively reasonable.

### C. Simulation

For the simulated experiments, we used the same object geometry models as in the real-life experiments. One or more objects were fixed to one position in space (considered "known"), and one or more objects were assigned variable poses (considered "unknown"). Because of the lack of noise, we were able to use stricter parameter settings: $\delta$ =2 cm, $d_0^+$ =0.7 cm, $d_0^-$ =0.4 cm and $b = 0.5$.

In accordance with the principles put forth in Section II-C, we then sampled the distribution of the On function by randomly selecting poses for the variable objects and evaluating the On function for each. The figures in this section each show 2500 samples that evaluated to On $> 0.5$. Note that the full 6 DOF pose was variable, although the figures only show the position of the COM.

Figure 6 contains the models of the objects used in the simulation: a square table, a cereal box and a roughly cubical larger box, being the same models used with actual objects above.

Two basic cases are shown in Figures 7 and 8. The former shows samples of $A$'s position, given that it is On the table; the latter, given that it is On $B$. The stratification that can be observed corresponds to $A$ standing up, and lying on its side or back, respectively. This arises directly from the On function and shows how On can encapsulate complex modes of configurations implicitly. Automatic clustering would allow for the extraction of these modes, which might then be used in high-level qualitative reasoning.

Two other configurations of the object $B$ are shown in Figure 9. These illustrate how the inclination of the support object is taken into account in the On computation. Not all points "above" $B$ are valued equally, as might be the case in a purely geometrical approach, but rather points corresponding to a largely vertical contact normal are considered more feasible. In the second image, the distribution is concentrated to a narrow region corresponding to $A$ balancing on the topmost edge of $B$ (which translates to a low On value in absolute terms, despite being the global maximum).

The potential uses of these distributions are many. As explained in Section II-C, they can be translated into probability distributions. In a search scenario, where it is known that $A$ On $B$, and $B$'s pose is known, the distribution may be used to direct the search for $A$. If the pose of $B$ is not known, the distribution (as computed by assuming $B$ *were* known) can be compared to an uninformed prior on $A$'s location, allowing the robot to decide whether it is worth it to search for $B$ first, or if it is better to look for A directly.

In that same vein, Figure 10 contains the result of a *chained* sampling, where both objects $A$ and $B$ were allowed to vary randomly. Only when both $B$ On Table and $A$ On $B$ were greater than $0.5$ was the position of $A$ plotted. In other words, what is represented is the distribution over $A$'s position, given that $A$ On $B$ and $B$ On Table, but with $B$'s exact pose unknown.

This type of chained inference allows for e.g. searching for $A$ without first locating $B$, while still utilizing the knowledge that $A$ On $B$. As stated above, the distribution can be compared to the prior of $A$, and $A$ given $A$ On $B$, to determine whether it is more beneficial to locate $B$ first or not.

## IV. Discussion

This work has only begun to explore the possible uses to a mobile robot of the conceptualization proposed. We would like to extend the work in several ways. First, evaluating the efficiency of object search utilising the results of this paper, as well as exploring how the principle of using functional criteria can be generalized through a similar treatment of

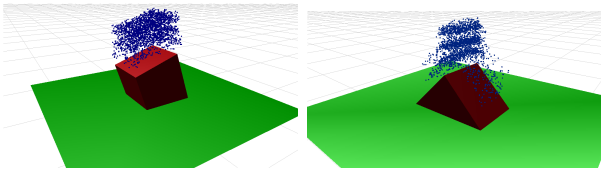


Fig. 9. Position of $A$, given "$A$ On $B$"

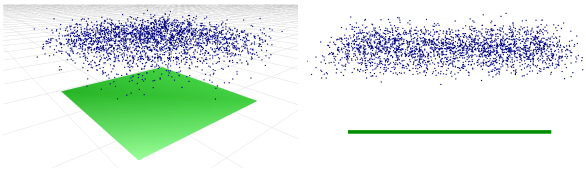


Fig. 10. Position of $A$, given "$A$ On $B$, $B$ On Table"

other important spatial relations, especially topological ones such as "in" or "at". In addition, we hope to integrate this conceptualization of "on" with information from language and the OMICS database, and conversely to use it to generate spatial utterances and generalizations about typical relations between objects. Furthermore our implementation has been limited to box and plane shapes; it will readily extend to any convex 3D shapes, but non-convex shapes require that further assumptions be made.

The perceptual model described in Section II-B assumes knowledge of the involved objects' geometry, poses and centers of mass. Whereas a human is able to estimate these quantites, even for novel objects, and/or extrapolate them based on experience, a robot may not always have access to good such estimates from its visual system. Vision is not the focus of this work, however, and the soft nature of the functions gives some robustness to poor visual information; moreover, and more importantly: as shown in Section II-D, when the relation information is used in the "opposite direction", such as in search, poses do not need to be provided.

The descriptors used in this work were selected in an *a priori* fashion, and the relevant weights were adjusted manually. In the future, we hope to achieve a more objective correspondence between the model and the idealized conceptualization of support – and other such idealized conceptualizations – through learning, either based on studies of human classification or on mechanical simulation. The choice of which features to use in the first place is a still more challenging learning goal, which must nevertheless be tackled in order to allow the approach to be applied to a far wider set of conceptualizations. A concept such as mechanical support cannot be acquired from scratch without learning from experience with manipulating physical bodies, connecting physical forces that are felt to visual properties of objects and the effects of actions. More work needs to be done on using such feedback to build functional spatial concepts, work that cannot be separated from the larger scope of imbuing robots with greater intelligence.

## V. Conclusions

We proposed an idealized cognitive model for the core concept underlying English "on", *viz.* mechanical support, in order to give us a functionally grounded abstraction primitive to use with qualitative reasoning and learning, top-down processes such as visual search, and linguistic interaction. A novel perceptual model was designed and implemented to approximately analyze real-world scenes in terms of this model, and results of experiments with real-world data were presented. Finally, we contributed a method to synthesize expectations about the metric location of an object to aid in e.g. efficient search.

## References


[1] Askoxford: Language facts. http://www.askoxford.com/oec/mainpage/oec02/, 2010.

[2] Openmind indoor commonsense. http://openmind.hri-us.com/, 2010.

[3] A.G. Cohn and S.M. Hazarika. Qualitative spatial representation and reasoning: An overview. *Fundamenta Informaticae*, 2001.

[4] Kenny Coventry and Simon Garrod. *Saying, seeing and acting : the psychological semantics of spatial prepositions*. Hove, 2003.

[5] Thomas David Garvey. *Perceptual strategies for purposive vision.* PhD thesis, Stanford, CA, USA, 1976.

[6] A. Herskovits. *Language and Spatial Cognition*. Cambridge University Press, 1986.

[7] G. Lakoff. *Women, fire and dangerous things: what categories reveal about the mind*. University of Chicago Press, 1987.

[8] S. Levinson. *Language and Space*, chapter Frames of Reference and Molyneux's question: cross-linguistic evidence. MIT Press, 1996.

[9] S. Levinson and S. Meira. 'natural concepts' in the spatial topological domain – adpositional meanings in crosslinguistic perspective: An exercise in semantic typology. *Language*, 79(3), 2003.

[10] K. Lockwood, K. Forbus, D.T. Halstead, and J. Usher. Automatic categorization of spatial prepositions. In *Proceedings of the 28 th Annual Conference of the Cognitive Science Society.*, 2006.

[11] J. O'Keefe. *The Spatial Prepositions*, chapter 7. The MIT Press, 1999.

[12] J. Piaget and B. Inhelder. *The Child's Conception of Space*. Routledge & Keagan Paul Ltd., 1956.

[13] T. Regier and L. A. Carlson. Grounding spatial language in perception: An empirical and computational investigation. *Journal of Experimental Psychology*, 130(2):273–2098, 2001.

[14] A. Richtsfeld, T. Mörwald, M. Zillich, and M. Vincze. Taking in shape: Detection and tracking of basic 3d shapes in a robotics context. In *Computer Vision Winder Workshop*, pages 91–98, 2010.

[15] L. Talmy. Force dynamics in language and cognition. *Cognitive Science*, 1988.

[16] John K. Tsotsos and Ksenia Shubina. Attention and visual search : Active robotic vision systems that search. In *International Conference on Computer Vision Systems ICVS'07*, page 539, Washington, DC, USA, 2007. IEEE Computer Society.

[17] Lambert E. Wixson and Dana H. Ballard. Using intermediate objects to improve the efficiency of visual search. *Int. J. Comput. Vision*, 12(2-3):209–230, 1994.

[18] Yiming Ye and John K. Tsotsos. Where to look next in 3d object search. In *ISCV '95: Proceedings of the International Symposium on Computer Vision*, page 539, Washington, DC, USA, 1995. IEEE Computer Society.

[19] Yiming Ye and John K. Tsotsos. Sensor planning for 3d object search. *Comput. Vis. Image Underst.*, 73(2):145–168, 1999.

# Simultaneous Object Class and Pose Estimation for Mobile Robotic Applications with Minimalistic Recognition

Alper Aydemir, Adrian N. Bishop and Patric Jensfelt

***Abstract*— In this paper we address the problem of simultaneous object class and pose estimation using nothing more than object class label measurements from a generic object classifier. We detail a method for designing a likelihood function over the robot configuration space. This function provides a likelihood measure of an object being of a certain class given that the robot (from some position) sees and recognizes an object as being of some (possibly different) class. Using this likelihood function in a recursive Bayesian framework allows us to achieve a kind of spatial averaging and determine the object pose (up to certain ambiguities to be made precise). We show how inter-class confusion from certain robot viewpoints can actually increase the ability to determine the object pose. Our approach is motivated by the idea of minimalistic sensing since we use only class label measurements albeit we attempt to estimate the object pose in addition to the class.**

## I. Introduction

Object search (or active visual (object) search) is an important component of a mobile robot's action space [1]. For example, finding, identifying and localizing the pose of objects is a prerequisite for a robot that wishes to interact with objects in the environment. In [2] it is shown that a human understanding of space is significantly based on the objects present in the scene.

In this paper we are interested in the problem of simultaneous object class and pose estimation using a generic object classifier and a spatially dependent measurement likelihood model. One novelty we claim is the ability to estimate both the class and pose of objects in the environment given only measurements of the object class. Indeed, we attempt to push the limits of what information can be estimated given nothing more than a simple object class return and a model of the spatial likelihood for that class return.

Most existing classification and pose estimation algorithms match local geometric features *of the object*, such as corners, edges, holes and surfaces to a precise geometric model *of the object* [3]–[7]. Such techniques require extensive storage and training data and are far from minimalistic. In addition, these approaches are sensitive to object occlusions etc where object class measurements are possible but precise geometrical measurements of the object are not possible.

Other techniques [6], [8], [9] use a large number of labeled images taken from different poses and attempt to match specific images in order to determine the pose. The accuracy of these approaches increases with the amount of training and reference images. This technique critically ignores the relative geometry of the sensor and the object and the affect of this relationship on the likelihood of recognizing certain views (or more generally whole objects). In particular, we show how we can achieve more with much less input if we consider this relationship explicitly.

The authors are with the Centre for Autonomous Systems (CAS) at the Royal Institute of Technology (KTH), Stockholm, Sweden. This work was supported by the Swedish Foundation for Strategic Research (SSF) through CAS and also via the EU FP7 project CogX.

*1) Original Contribution:* We differ from existing object search methods in the design of our spatial likelihood functions. We will highlight throughout the paper that one novelty of our approach is that it is truly minimalistic in nature. By measuring only the object class label we attempt to extract both the true object class and object pose (orientation and location). Indeed, we generally ignore the notion of object view recognition and assign class labels only to entire objects. We certainly ignore any geometrical aspects of the object and we employ generic classifiers (we ignore the particular features employed by the classifier and indeed in our experiments we use a recognition algorithm from the literature which does not make use of any geometrical model of the object). We can extract certain estimates of the object pose purely from the structure of the likelihood functions which are defined over the robot configuration space and hence are geometrically related to the object pose. In addition, we show how inter-class confusion, e.g. the ability to mistakenly measure multiple class labels for a particular object type from certain views, can be advantageous to the estimation problem (specifically the estimation of object pose). We can achieve a high degree of accuracy in pose estimation using our technique and exploiting inter-class confusion. As far as we are aware our technique is novel and truly minimalistic.

*2) Paper Outline:* This paper is organized as follows. In the next section we outline the general notation used throughout the paper along with the basics of the robot dynamic model considered. In Section III we formulate the problem and outline the design of the likelihood function. We also provide some intuition regarding the design of the spatial likelihood function through example. Furthermore, in Section III we outline the recursive Bayesian algorithm for computing an objects pose and class and we highlight the algorithm behaviour using a simple toy example. We show how measurements of the class label alone can be used to determine accurately the object pose given a suitable likelihood function defined over the robot configuration space. We then outline an extension of the algorithm in Section IV for object class and orientation estimation over a grid. In Section V we provide the results of a practical experiment over a grid and in Section VI we discuss the results and directions for future work. Our conclusion is given in Section VII.

## II. PRELIMINARIES

In this section we outline some notational preliminaries and the robot dynamical model considered.

### A. Notation

Introduce a global coordinate frame $\mathcal{C}$ at some pre-defined time $t_0$. Consider a set of *objects* $\mathcal{O} = \{o_1, \ldots, o_{n_o}\}$ with position $\mathbf{x}_i \in \mathbb{R}^2$ and orientation $\phi_i \in \mathrm{S}^1$. Consider an arbitrary object $o_i$ placed so that the $\mathbb{R}^2$ location of the object's center hovers over the origin of $\mathcal{C}$ at $t_0$. Introduce a local two-dimensional coordinate frame $\mathcal{C}_i$ at the center of $o_i$. Then the orientation $\phi_i$ is defined as the relative rotation of $\mathcal{C}_i$ with respect to $\mathcal{C}$. Each object $o_i$ belongs to a class $\{c_j\}_{j=1}^{n_c}$ or the *unclassified* or *non-object* class $c_0$.

The position of a single mobile robot is denoted by $\mathbf{s} \in \mathbb{R}^2$ with heading $\theta \in \mathrm{S}^1$. The distance between the robot and $o_i$ is given by $r_i = \|\mathbf{x}_i - \mathbf{s}\|$. The relative direction to $o_i$ from the robot's heading is given by $\vartheta_i = \alpha_i - \theta$ where $\alpha_i$ is the azimuthal bearing to $o_i$ in the global coordinate system and $\vartheta_i \in \mathrm{S}^1$. We then define a viewpoint $\mathbf{p}_i = [\mathbf{s}\ \vartheta_i]^\top$.

### B. Dynamics

Introduce the matrix Lie group $\mathbb{SE}(2)$ with group element $\mathbf{X}(\psi, \mathbf{q}) \in \mathbb{SE}(2)$ with $\mathbf{q} = [x\ y]^\top \in \mathbb{R}^2$ and a group (matrix) multiplication operator. An element $\mathbf{X}(\theta, \mathbf{r}) \in \mathbb{SE}(2)$ acts on a point $\mathbf{p}_i \in \mathbb{R}^2$ by mapping it to $(\mathbf{R}(\theta)\mathbf{p}_i + \mathbf{r}) \in \mathbb{R}^2$. Here $(\mathbf{R}(\theta)$ is the rotation matrix defined as

$$\mathbf{R}(\psi) = \begin{bmatrix} \cos\psi & -\sin\psi \\ \sin\psi & \cos\psi \end{bmatrix} \tag{1}$$

and note that all elements in $\mathbb{SO}(2)$ are congruent to such a matrix. For notational brevity we write the action of $\mathbf{X}(\psi, \mathbf{r})$ on $\mathbf{q}$ as

$$\mathbf{X}(\psi, \mathbf{r}) \circ \mathbf{q} = \mathbf{R}(\psi)\mathbf{q} + \mathbf{r} \tag{2}$$

which constitutes a left action of $\mathbb{SE}(2)$ on $\mathbb{R}^2$. The inverse $\mathbf{X}^{-1}(\theta, \mathbf{r}) \in \mathbb{SE}(2)$ maps $\mathbf{p}_i$ to $\mathbf{R}^\top(\theta)\mathbf{p}_i - \mathbf{R}^\top(\theta)\mathbf{r}$ and the identity is given by $\mathbf{X}(0, \mathbf{0}) \in \mathbb{SE}(2)$.

Associated with $\mathbb{SE}(2)$ is the vector space $\mathrm{se}(2)$ which is a Lie algebra with respect to the Lie bracket operation. We define the basis of $\mathrm{se}(2)$ by $\{\mathbf{E}_x, \mathbf{E}_y, \mathbf{E}_\psi\}$ with

$$\mathbf{E}_i = \begin{bmatrix} 0 & 0 & 1(i=x) \\ 0 & 0 & 1(i=y) \\ 0 & 0 & 0 \end{bmatrix},\ \mathbf{E}_\theta = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{3}$$

with $i \in \{x, y\}$ and where $1(\cdot)$ is an indicator function.

Given translational and angular velocity control inputs $u_1 = v$ and $u_2 = \omega$ we then have

$$\dot{\mathbf{X}}(\theta, \mathbf{t}) = \mathbf{X}(\theta, \mathbf{t})\left(\mathbf{E}_x u_1 + \mathbf{E}_\theta u_2\right) \tag{4}$$

which constitutes a left-invariant, drift-free system on the group $\mathbb{SE}(2)$. This model is the Lie group representation of the unicycle model and is our robot kinematic model.

## III. PROBLEM FORMULATION

In this section we outline the probabilistic framework within which our estimation problem is formulated.

### A. Classification Likelihoods on Lie Groups

For each $\mathbf{p}(t)$ the robot takes measurements of the potential class of $o_i$ in the form

$$\mathbf{y}_i(t) = [\widehat{c}_j\ \ldots\ \widehat{c}_k]^\top \tag{5}$$

with $\mathbf{y} = [\mathbf{y}_1^\top\ \ldots\ \mathbf{y}_{n_o}^\top]^\top$. This means that a measurement of object $o_i$ can return more than a single class value[1].

We model the likelihood of measuring $\widehat{c}_j$ for $o_i$ as a function of the robot pose. In fact, we model this likelihood as a sum of Gaussian densities on the Lie group $\mathbb{SE}(2)$. Consider an arbitrary normal density in $\mathbb{R}^n$ of the form

$$\gamma(\mathbf{x} - \boldsymbol{\mu}, \boldsymbol{\Sigma}) = \frac{1}{(2\pi)^{\frac{n}{2}}|\boldsymbol{\Sigma}|^{1/2}} \exp\left(\frac{1}{2}\|\boldsymbol{\Sigma}^{\frac{-1}{2}}(\mathbf{x} - \boldsymbol{\mu})\|_2^2\right) \tag{6}$$

where $\boldsymbol{\Sigma}$ is the covariance matrix and $\boldsymbol{\mu}$ is the mean. A Gaussian distribution on the Lie group $\mathbb{SO}(2)$ is given by

$$\chi(x - \mu, \sigma^2) = \frac{1}{\sigma\sqrt{2\pi}} \sum_{k \in \mathbb{Z}} \exp\left[\frac{-(x - \mu - 2\pi k)^2}{2\sigma^2}\right] \tag{7}$$

and if $\sigma^2 << 2\pi$ in $\chi(x - \mu, \sigma^2)$ then $\chi(x - \mu, \sigma^2)$ can be approximated well by the case $k = 0$ in (7). A Gaussian on the product space $\mathbb{SE}(2)$ can then be denoted by $\zeta(\mathbf{x} - \boldsymbol{\mu}, \boldsymbol{\Sigma})$. We state the following lemma for completeness.

*Lemma 1 ( [10]):* There exists an integer $m$ and constants $w_i > 0$ with $\sum_{i=1}^m w_i = 1$, such that the Gaussian sum

$$p_{approx}(\mathbf{x}) = \sum_{i=1}^{m} w_i \gamma(\mathbf{x} - \boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i) \tag{8}$$

can approximate any density function $p(\mathbf{x})$ as closely as desired in the sense that $\int_{\mathbb{R}^n} |p(\mathbf{x}) - p_{approx}(\mathbf{x})|\ d\mathbf{x}$ can be made arbitrarily small.

Recall that the element $\mathbf{X}(\psi, \mathbf{r}) : \mathbb{R}^2 \to \mathbb{R}^2$ acts on points via left translation denoted by $\mathbf{X}(\psi, \mathbf{r}) \circ \mathbf{q}$. For notational brevity we introduce the following notational definition

$$\mathbf{X}(\psi, \mathbf{r}) \circ [\mathbf{q}\ q_3\ q_4\ \ldots\ q_n]^\top = [\mathbf{R}(\psi)\mathbf{q} + \mathbf{r}\ q_3\ q_4\ \ldots\ q_n]^\top \tag{9}$$

which means that $\mathbf{X}(\psi, \mathbf{r}) : \mathbb{R}^2 \times \mathcal{A} \to \mathbb{R}^2 \times \mathcal{A}$ by acting on the first two dimensions in the standard way and leaving the remaining $n - 2$ dimensions unchanged.

We model the likelihood function by

$$p(\widehat{c}_i, o_j | c_i, \phi_j, \mathbf{x}_j) = \mathrm{P}(\widehat{c}_i, o_j | c_i) \sum_{k_i} \frac{w_{k_i}}{(2\pi)^{\frac{3}{2}}|\boldsymbol{\Sigma}_{k_i}|^{1/2}} \times \exp\left(\frac{1}{2}\|\boldsymbol{\Sigma}_{k_i}^{\frac{-1}{2}}(\mathbf{p}_j - \mathbf{X}(\phi_j, \mathbf{x}_j) \circ \mathbf{q}_{k_i})\|_2^2\right) \tag{10}$$

where $\sum_{k_i} w_{k_i} = 1$ with $i = \{1, \ldots, n_c\}$. For an object with position $\mathbf{x}_j$ and orientation $\phi_j$ we define $\mathbf{X}(\phi_j, \mathbf{x}_j)\mathbf{q}_{k_i}$

[1]For example, observing a car from the front may yield several positive car model class returns). We assume perfect data association, i.e. we know which objects $o_i$ generate particular class measurements.

with $\mathbf{X}(\phi_j, \mathbf{x}_j) \in \mathbb{SE}(2)$ and $\mathbf{q}_{k_i} \in \mathbb{R}^2 \times \mathbb{SO}(2)$ as the *mean* of the $k_i{}^{th}$ Gaussian and $\mathbf{\Sigma}_{k_i}$ is the *covariance*[2].

The term $\mathrm{P}(\widehat{c}_i, o_j | c_i)$ specifically deals with the likelihood of $o_j$ being of class $c_i$ given the measurement $\widehat{c}_i$ whereas $p(\widehat{c}_i, o_j | c_i, \phi_j, \mathbf{x}_j)/\mathrm{P}(\widehat{c}_i, o_j | c_i)$ is the likelihood of $o_j$ being in position $\mathbf{x}_j$ with orientation $\phi_j$.

The likelihood $p(\widehat{c}_i, o_j | c_i, \phi_j, \mathbf{x}_j)$ is also a probability density function such that for bounded regions of $\mathcal{A}$ of $\mathbb{SE}(2)$ with positive Lebesgue measure the integral

$$\int_{\mathcal{A}} p(\widehat{c}_i, o_j | c_i, \phi_j, \mathbf{x}_j) \; d\mathbf{p} \tag{11}$$

gives the probability of measuring $\widehat{c}_i$ for $o_j$ given that $o_k$ is of class $c_i$ and with orientation $\phi_j$ and position $\mathbf{x}_j$. We state this explicitly since we will require that the so-called *confusion densities* $p(\widehat{c}_j, o_k | c_i, \phi_k, \mathbf{x}_k)$ with $i \neq j$ satisfy the inequality

$$\int_{\mathcal{A}} p(\widehat{c}_j, o_k | c_i, \phi_k, \mathbf{x}_k) \; d\mathbf{p} \leq \int_{\mathcal{A}} p(\widehat{c}_i, o_k | c_i, \phi_k, \mathbf{x}_k) \; d\mathbf{p} \tag{12}$$

or $p(\widehat{c}_j, o_k | c_i, \phi_k, \mathbf{x}_k) \leq p(\widehat{c}_i, o_k | c_i, \phi_k, \mathbf{x}_k)$ for all bounded subsets $\mathcal{A}$ of $\mathbb{SE}(2)$ in a defined region of interest $\mathcal{R} \subset \mathbb{SE}(2)$. That is, over any bounded region in $\mathcal{R}$ we want the probability of measuring $\widehat{c}_j$ for $o_i$ to be less than (or equal to) the probability of measuring $c_i$ given that the object is of true class $c_i$ (and for all object poses).

Of course, this inequality cannot be satisfied over all of $\mathbb{SE}(2)$ if the confusion likelihood is also required to be a true density function. But to make this definition consistent we note that when viewed as a likelihood function $p(\widehat{c}_i, o_j | c_k, \phi_j, \mathbf{x}_j)$ is valid as long as it is congruent to a probability density function via multiplication by a constant.

We model the likelihood function of false positives by the following Gaussian mixture

$$p(\widehat{c}_i, o_j | c_k, \phi_j, \mathbf{x}_j) = \textstyle\sum_{k_{ik}} \frac{\mathtt{common}(k_i, k_k) \; w_{k_{ik}}}{(2\pi)^{\frac{3}{2}} |\mathbf{\Sigma}_{k_{ik}}|^{1/2}} \times$$
$$\exp\left( \tfrac{1}{2} \| \mathbf{\Sigma}_{k_{ik}}^{\frac{-1}{2}} \left( \mathbf{p} - \mathbf{X}(\phi_j, \mathbf{x}_j) \circ \mathbf{q}_{k_{ik}} \right) \|_2^2 \right) \mathrm{P}(\widehat{c}_i, o_j | c_k) \tag{13}$$

where the sum over $k_{ik}$ has at most $\min(k_i, k_k)$ terms and (12) must hold in $\mathcal{R} \subset \mathbb{SE}(2)$. The function

$$\mathtt{common}(k_i, k_k) \in \{0, 1\} \tag{14}$$

captures the fact that an object $o_j$ can be confusingly observed as $\widehat{c}_i$ and/or $\widehat{c}_k$ from some robot positions because the underlying true classes $c_i$ and $c_k$ share a *common* indistinguishability from such locations[3].

The class $c_0$ is used to model unclassified classes or locations in space where no object exists. The likelihood $p(\widehat{c}_i, o_j | c_0, \phi_j, \mathbf{x}_j)$ where $i \neq 0$ is given by

$$p(\widehat{c}_i, o_j | c_0, \phi_j, \mathbf{x}_j) = \mathrm{P}(\widehat{c}_i, o_j | c_0) \tag{15}$$

which although not a true likelihood function is valid over any bounded region $\mathcal{R} \subset \mathbb{SE}(2)$ since it is congruent to a uniform density over $\mathcal{R}$. For all classes for which it is defined we now require $\sum_i \mathrm{P}(\widehat{c}_i, o_j | c_k) = 1$.

For much of the space $\mathbb{SE}(2)$ the object recognizer will not return any class value for $o_j$. We can (if desired) model the absence of any returns in $\{c_1, \ldots, c_{n_c}\}$ as a measurement of the dummy class $c_0$. We would then need to construct the likelihood $p(\widehat{c}_0, o_j | c_i, \phi_j, \mathbf{x}_j)$. We do not explore the design of this likelihood in detail since we will not (in our implementations) incorporate dummy measurements when no class is detected[4].

If we define $\mathbf{c} = [c_0 \; c_1 \; \ldots \; c_{n_c}]$ then the likelihood

$$p(\widehat{c}_i, o_j | \mathbf{c}, \phi_j, \mathbf{x}_j) = \sum_k p(\widehat{c}_i, o_j | c_k, \phi_j, \mathbf{x}_j) \tag{16}$$

is the multi-dimensional likelihood function of the object being in all of the defined classes and all poses given a particular class return. Given a return measurement $\mathbf{y}_i(t) = [\widehat{c}_a \; \widehat{c}_b \; \ldots \; \widehat{c}_z]^\top$ for object $i$ then the joint likelihood is

$$p(\mathbf{y}_j, o_j | \mathbf{c}, \phi_j, \mathbf{x}_j) = \prod_{\widehat{c}_k \in \mathbf{y}_j} p(\widehat{c}_k, o_j | \mathbf{c}, \phi_j, \mathbf{x}_j) \tag{17}$$

under a naive Bayesian assumption, i.e. under the assumption that $p(\widehat{c}_k, \cdot | \mathbf{c}, \cdot, \widehat{c}_j) = p(\widehat{c}_k, \cdot | \mathbf{c}, \cdot)$.

### *B. Example Likelihood Functions*

We now provide some intuition regarding the design of the likelihood functions. These examples are simplified but illustrate the heuristics behind the likelihood structure.

Consider an object $o_1$ of class $c_1$ located at the origin at time $t_0$ with defined orientation $\phi_1 = 0$. An object classifier is trained on object $o_1$ from a number of relative positions denoted by $\mathbf{q}_{k_1}$ with $\mathbf{q}_{k_1} = [q_{k_1}^1 \; q_{k_1}^2 \; 0]^\top$. For the $k^{th}$ training position we define a Gaussian $\zeta(\mathbf{p}_1 - \mathbf{X}(\phi_1, \mathbf{x}_1) \circ \mathbf{q}_{k_1}, \mathbf{\Sigma}_{k_1})$ where $\mathbf{\Sigma}_{k_1}$ is tuned based on the specific classifiers properties[5]. We define $p(\widehat{c}_1, o_i | c_1, \phi_i, \mathbf{x}_i)$ as the sum of such Gaussians as in (10) with $w_{k_1} = 1/4$ and $\mathrm{P}(\widehat{c}_1, o_1 | c_1) = 1$. In this example we set $\mathbf{q}_{1_1} = [10 \; 0 \; 0]^\top$, $\mathbf{q}_{2_1} = [-10 \; 0 \; 0]^\top$, $\mathbf{q}_{3_1} = [0 \; 10 \; 0]^\top$ and $\mathbf{q}_{4_1} = [0 \; -10 \; 0]^\top$ with $\mathbf{\Sigma}_{k_1} = \mathrm{diag}(10, 10, \pi/4)$. Now consider a random object $o_i$ at $\mathbf{x}_i =$

[2]We note at this point that the Gaussian parameters $\mathbf{q}_{k_i}$ and $\mathbf{\Sigma}_{k_i}$ are defined based on the training scheme of the object classifier and $\mathbf{p}_j$ is the robot position when the relevant class labels are measured. Heuristically, $\mathbf{q}_{k_i}$ is taken to be (one of) the sensor's position in $\mathbb{SE}(2)$ at training time relative to the object (which is located at the origin during training with the reference orientation). The variance is (in this paper) tuned to provide a realistic model of the spatial dependence of the recognition algorithm at run time to the trained classifier models. In the next subsection we provide an example further illustrating how the likelihood functions are created. However, we note here that the motivation for these likelihood functions is motivated from experience where we have noticed that often simply by measuring the class label for an entire object (not view point) we *most likely* restricted one of a small number of points. In reverse, given a known robot position, the object is *most likely* in one of a small number of locations with one of a small number of orientations.

[3]Such a case happens, for example, if there are object types which look very similar or share a similar internal representation (ambiguous objects or object-data) from certain views.

[4]The reason we do not generate dummy measurements is that $\widehat{c}_0$ will, in general, provide little information about the true class $c_j$ (including $c_0$) for most robot positions $\mathbf{p}$ and objects $o_i$. Heuristically, over bounded regions the likelihood $p(\widehat{c}_0, o_j | c_i, \phi_j, \mathbf{x}_j)$ would resemble a constant minus the sum (16). Of course this would require some further justification in order for $p(\widehat{c}_0, o_j | c_i, \phi_j, \mathbf{x}_j)$ to be valid as a likelihood function.

[5]We discuss later that an interesting direction for future work is the design of reinforcement learning schemes for tuning such parameters.

$[5\ 5]^\top$ with $\phi_i = 45^o$. We plot $p(\widehat{c}_1, o_i|c_1, \phi_i, \mathbf{x}_i)$ with $\vartheta_i = 0$ over $\mathbf{s} \in \mathbb{R}^2$ in Figure 1.

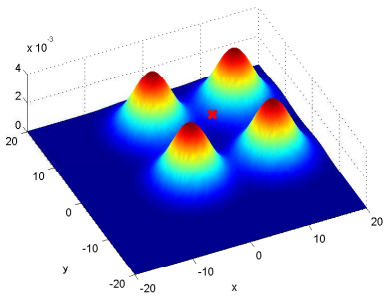



Fig. 1. An example likelihood function with $\vartheta_i = 0$ over $\mathbf{s} \in \mathbb{R}^2$.

Figure 1 shows from which positions in space relative to the target object $o_i$ the likelihood of the class of $o_i$ being $c_1$ is given that we measure $\widehat{c}_1$ (and in essence assume robot relative heading $\vartheta_i$ invariance - e.g. this will hold for omni-directional cameras). Note that the likelihood just demonstrated is symmetric in terms of the orientation $\phi_i$, and as a result $\phi_i$ can be determined only up to rotations modulo $\pi/2$ given measurements of the class $\widehat{c}_1$. We outline the specific estimation technique in the next subsection and later provide examples of the pose accuracy that can be achieved.

However, we now demonstrate how inter-class confusion can be aid the pose estimation problem if the views from which the confusion is likely are not maximal. Consider an object $o_2$ of class $c_2$ located at the origin at time $t_0$ with defined pose $\phi_2 = 0$. An object classifier is trained on object $o_2$ from a number of relative positions denoted by $\mathbf{q}_{k_2}$ and for the $k^{th}$ position we define a Gaussian $\zeta(\mathbf{p}_2 - \mathbf{X}(\phi_2, \mathbf{x}_2) \circ \mathbf{q}_{k_2}, \mathbf{\Sigma}_{k_2})$. Then we define $p(\widehat{c}_2, o_i|c_2, \phi_i, \mathbf{x}_i)$ as the sum of such Gaussians as in (10) with $w_{k_2} = 1/4$ and $\mathrm{P}(\widehat{c}_2|c_2) = 1$. In this example we set $\mathbf{q}_{i_2} = \mathbf{q}_{i_1}$ and $\mathbf{\Sigma}_{k_2} = \mathbf{\Sigma}_{k_1}$. Now consider a random object $o_i$ at $\mathbf{x}_i = [5\ 5]^\top$ with $\phi_i = 45^o$. The plot of $p(\widehat{c}_2, o_i|c_2, \phi_i, \mathbf{x}_i)$ with $\vartheta_i = 0$ over $\mathbf{s} \in \mathbb{R}^2$ is identical in this case to the likelihood shown in Figure 1.

Now suppose from a number of positions we know that $o_1$ and $o_2$ can be confusingly recognized as both $c_1$ and $c_2$ in some instances. In this example,

$$\mathtt{common}(2_1, 2_2) = 1 \tag{18}$$

and all other $\mathtt{common}(\cdot)$ equal to zero. We let $\mathbf{\Sigma}_{k_{1,2}} = \mathbf{\Sigma}_{k_2} = \mathbf{\Sigma}_{k_1}$ and $w_{k_{1,2}} = 1/4$. Also let $\mathrm{P}(\widehat{c}_2|c_1) = \mathrm{P}(\widehat{c}_1|c_2) = 1/2$ and now let $\mathrm{P}(\widehat{c}_i|c_i) = 1/2$. Then $p(\widehat{c}_j, o_i|c_j, \phi_i, \mathbf{x}_i)$, for $j = \{1, 2\}$, with $\vartheta_i = 0$ over $\mathbf{s} \in \mathbb{R}^2$ is the same shape as in Figure 1 except both likelihoods are weighted by $1/2$. In Figure 2 we plot $p(\widehat{c}_1, o_i|c_2, \phi_i, \mathbf{x}_i) = p(\widehat{c}_2, o_i|c_1, \phi_i, \mathbf{x}_i)$ with $\vartheta_i = 0$ over $\mathbf{s} \in \mathbb{R}^2$ for the same random object $o_i$ at $\mathbf{x}_i = [5\ 5]^\top$ with $\phi_i = 45^o$.

Given a random object $o_i$ of class $c_1$ or $c_2$ we now gain some intuition about how class confusion can aid in removing any ambiguity regarding the pose of the object. For example, if $o_i$ were viewed from a number of robot positions around $\mathbf{X}(\phi_i, \mathbf{x}_i) \circ \mathbf{q}_{2_1}$, i.e. around the confusion

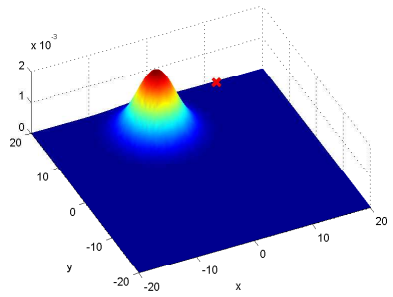



Fig. 2. A confusion likelihood function with $\vartheta_i = 0$ over $\mathbf{s} \in \mathbb{R}^2$.

peak, and both $\widehat{c}_1$ and $\widehat{c}_2$ were measured at these positions then the likelihood of the object pose would be (significantly) dominated by a single mode at the true pose. We will explore a detailed toy example illustrating this property later. We will also explore a practical example showing a real-world experimental result.

### C. Maximum A Posterior Probabilities

In terms of Bayes' rule we know

$$p(c_i, \phi_j, \mathbf{x}_j|\widehat{c}_i, o_j) = \frac{p(\widehat{c}_i, o_j|\mathbf{c}, \phi_j, \mathbf{x}_j)p(c_i, \phi_j, \mathbf{x}_j|o_j)}{p(\widehat{c}_i, o_j)} \tag{19}$$

or in terms of $\mathbf{y}(t)$ we have

$$p(c_i, \phi_j, \mathbf{x}_j|\mathbf{y}_j, o_j) = \frac{p(\mathbf{y}_j, o_j|\mathbf{c}, \phi_j, \mathbf{x}_j)p(c_i, \phi_j, \mathbf{x}_j|o_j)}{p(\mathbf{y}_j, o_j)} \tag{20}$$

where the denominator is given by

$$p(\mathbf{y}_j, o_j) = \int_{\mathbb{SE}(2)} p(\mathbf{y}_j, o_j|\mathbf{c}, \phi_j, \mathbf{x}_j)p(c_i, \phi_j, \mathbf{x}_j|o_j)\ d\phi_j d\mathbf{x}_j \tag{21}$$

Note we have neglected illustrating the dependence on time but the recursion is clear with the prior $p(c_i, \phi_j, \mathbf{x}_j)$ at time $t$ equal to the posterior $p(c_i, \phi_j, \mathbf{x}_j|\mathbf{y}_j, o_j)$ computed at some time $\tau < t$. We also know that

$$\mathrm{P}(c_i|\mathbf{y}_j, o_j) = \int_{\mathbb{SE}(2)} p(c_i, \phi_j, \mathbf{x}_j|\mathbf{y}_j, o_j)\ d\phi_j d\mathbf{x}_j \tag{22}$$

is the posterior probability of object $o_j$ being of class $c_i$ given the measurements $\mathbf{y}_j$ and $\sum_i \mathrm{P}(c_i|\mathbf{y}_j, o_j) = 1$ where $i = 0$ can be included naturally. For any $o_j$ we then have

$$\sum_i \int_{\mathbb{SE}(2)} p(c_i, \phi_j, \mathbf{x}_j|\mathbf{y}_j, o_j)\ d\phi_j d\mathbf{x}_j = 1 \tag{23}$$

where the sum is taken over all classes $c_0$ to $c_{n_c}$.

If we want the maximum a posterior (MAP) class and object orientation (or pose) then we can take the maximum class index and object pose estimates via

$$\{\widetilde{c}_i, \widetilde{\phi}_j, \widetilde{\mathbf{x}}_j\} = \underset{i, \phi_j, \mathbf{x}_j}{\mathrm{argmax}} \{p(c_i, \phi_j, \mathbf{x}_j|\mathbf{y}_j, o_j)\}_{i \in \{1, \ldots, n_c\}} \tag{24}$$

where $\widetilde{c}_i$ is the MAP class estimate for object $j$ corresponding to the maximization argument index $i$.

In general, (24) leads to $n_c$ maximization problems for each $o_j$. Each density is often multi-modal but each

mode can be determined easy via grid-search. If $\widehat{c}_k \notin \mathbf{y}_j(t)$ and $p(\widehat{c}_k, o_j|c_i, \phi_j, \mathbf{x}_j) = 0$ for all $i \neq k$ then $p(c_i, \phi_j, \mathbf{x}_j|\mathbf{y}_j, o_j) = 0$ at time $t$ and at least one maximization problem is avoided.

### D. Bringing it All Together with a Toy Example

A toy example is now examined in order to further develop an intuition regarding the approach outlined in this paper. A more detailed practical experiment is given later in the paper.

The fact we can localize the pose of the object accurately (even up to an ambiguity determined by the number of Gaussians in the likelihood function) is quite novel given we only use class label measurements. However, we go further then this and show how class confusions (from certain view points) can even reduce the number of ambiguities.

Consider an object $o_1$ located at $\mathbf{x}_1 = [5\ 5]^\top$ with true orientation $\phi_1 = 45^o$. Consider two potential object classes $c_1$ and $c_2$ with defined likelihood functions

$$p(\widehat{c}_i, o_1|c_i, \phi_1, \mathbf{x}_1) = \frac{0.2495}{(2\pi)^{\frac{3}{2}}|\mathbf{\Sigma}_{1_i}|^{1/2}} \times \exp\left(\frac{-1}{2}\|\mathbf{\Sigma}_{1_i}^{\frac{-1}{2}}\left(\mathbf{p} - \mathbf{X}(\phi_1, \mathbf{x}_1)\mathbf{q}_{1_i}\right)\|_2^2\right) + \frac{0.2495}{(2\pi)^{\frac{3}{2}}|\mathbf{\Sigma}_{2_i}|^{1/2}} \exp\left(\frac{-1}{2}\|\mathbf{\Sigma}_{2_i}^{\frac{-1}{2}}\left(\mathbf{p} - \mathbf{X}(\phi_1, \mathbf{x}_1)\mathbf{q}_{2_i}\right)\|_2^2\right) \quad (25)$$

for both $i = 1$ and $i = 2$ (with $\mathrm{P}(\widehat{c}_i|c_i) = 1/2 - 0.001$ as a consequence). The mean parameters are given by $\mathbf{q}_{1_1} = \mathbf{q}_{1_2} = [0\ 10\ 0]^\top$ and $\mathbf{q}_{2_1} = \mathbf{q}_{2_2} = [0\ -10\ 0]^\top$. The false positive likelihoods are given by

$$p(\widehat{c}_i, o_1|c_j, \phi_1, \mathbf{x}_1) = \frac{0.2495}{(2\pi)^{\frac{3}{2}}|\mathbf{\Sigma}_{1_{i,j}}|^{1/2}} \times \exp\left(\frac{-1}{2}\|\mathbf{\Sigma}_{1_{i,j}}^{\frac{-1}{2}}\left(\mathbf{p} - \mathbf{X}(\phi_1, \mathbf{x}_1)\mathbf{q}_{1_{i,j}}\right)\|_2^2\right) \quad (26)$$

with $i \neq j \in \{1, 2\}$ and $\mathbf{q}_{1_{i,j}} = [0\ 10\ 0]^\top$ (and $\mathrm{P}(\widehat{c}_j|c_i) = 1/2 - 0.001$). The variance is given by $\mathbf{\Sigma}_{i_j} = \mathbf{\Sigma}_{1_{i,j}} = \mathrm{diag}(10, 10, \pi/4)$ for all combinations of $i$ and $j$. Now consider the class $c_0$ with

$$p(\widehat{c}_i, o_1|c_0, \phi_1, \mathbf{x}_1) = \mathrm{P}(\widehat{c}_i|c_0) = 0.001 \quad (27)$$

for $i \in \{1, 2\}$. The recognition system can return class measurements $\widehat{c}_1$ and $\widehat{c}_2$.

We plot $p(\widehat{c}_i, o_1|c_i, \phi_1, \mathbf{x}_1)$ and $p(\widehat{c}_i, o_1|c_j, \phi_1, \mathbf{x}_1)$ with $i \neq j \in \{1, 2\}$ and with $\vartheta_i = 0$ over $\mathbf{s} \in \mathbb{R}^2$ in Figure 3.

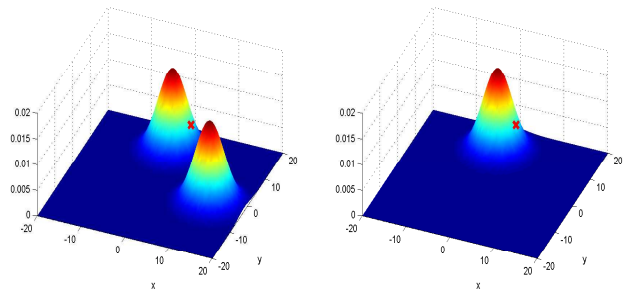


Fig. 3. The likelihoods $p(\widehat{c}_i, o_1|c_i, \phi_1, \mathbf{x}_1)$ and $p(\widehat{c}_i, o_1|c_j, \phi_1, \mathbf{x}_1)$ with $i \neq j \in \{1, 2\}$ evaluated at $\mathbf{x}_1 = [5\ 5]^\top$ and $\phi_1 = 45^o$. This shows the relationship between the robot position and the likelihoods.

In this example we assume $\mathbf{x}_1$ is known but the true object class and orientation $\phi_i$ is unknown. This is a reasonable approximation in many active object search problems[6]. In the next section, we consider a grid-based object search and orientation estimation problem where this assumption is explicitly realized.

The initial priors are thus $p(c_i, \phi_1, \mathbf{x}_1|o_1) = 1/(2\pi)$. We simulate measurements at a number of positions in space in order to examine their affect on the posterior densities.

`Time 1`

The robot position is given by $\mathbf{p} = \mathbf{X}(\phi_1, \mathbf{x}_1) \circ [2\ -10\ 0]^\top$. The measurements are given by $\mathbf{y}_1(1) = [\widehat{c}_1]^\top$. The posterior density functions are shown in Figure 4.

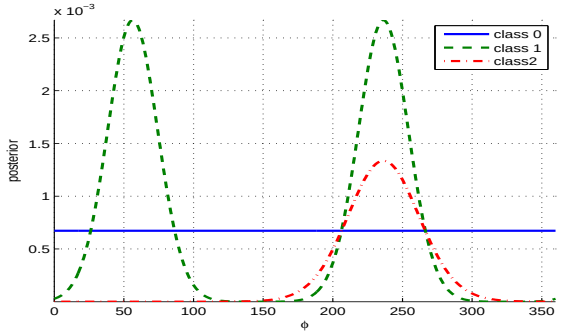



Fig. 4. The posteriors densities after the first measurement.

We know $\mathrm{P}(c_i|\mathbf{y}_j, o_j) = \int_{\mathbb{SO}(2)} p(c_i, \phi_j, \mathbf{x}_j|\mathbf{y}_j, o_j)\ d\phi_j$ and we can compute $\mathrm{P}(c_0|\mathbf{y}_1, o_1) = 0.4231$, $\mathrm{P}(c_1|\mathbf{y}_1, o_1) = 0.4244$ and $\mathrm{P}(c_2|\mathbf{y}_1, o_1) = 0.1524$ all at time 1. Then for the maximum class posterior estimate $\widetilde{c}_1$ we can compute the maximum $\mathrm{argmax}_{\phi_1} p(c_1, \phi_1, \mathbf{x}_1|\mathbf{y}_1(1), o_1)$ which is clearly (up to numerical tolerance) ambiguous with $\widetilde{\phi}_1 \approx 55^o$ and $\widetilde{\phi}_1 \approx 235^o$. The estimate of $\widetilde{c}_1$ is not overwhelmingly probable and the orientation estimate $\widetilde{\phi}_1$ is not exceedingly accurate since we have only employed a single measurement.

`Time 2`

The robot is at $\mathbf{p} = \mathbf{X}(\phi_1, \mathbf{x}_1) \circ [-2\ -10\ 0]^\top$. The measurements are given by $\mathbf{y}_1(2) = [\widehat{c}_1]^\top$. The posterior density functions are shown in Figure 5.

We compute $\mathrm{P}(c_0|\mathbf{y}_1, o_1) = 0.3019$, $\mathrm{P}(c_1|\mathbf{y}_1, o_1) = 0.5735$ and $\mathrm{P}(c_2|\mathbf{y}_1, o_1) = 0.1247$ at time 2. Then for the maximum class posterior estimate $\widetilde{c}_1$ we compute the maximum $\mathrm{argmax}_{\phi_1} p(c_1, \phi_1, \mathbf{x}_1|\mathbf{y}_1(1), o_1)$ which is again (up to numerical tolerance) ambiguous with $\widetilde{\phi}_1 \approx 45^o$ and $\widetilde{\phi}_1 \approx 225^o$. However, now the orientation estimate is accurate up to the ambiguity. The increased accuracy in the orientation (neglecting the ambiguity) is a result of the spatial averaging that occurs when observing the object from different robot positions (and this accuracy is quite interesting given we only physically measure the class label).

[6]For example, laser or stereo vision can be used to position objects in space in some scenarios but does not necessarily aid in the estimation of object class or orientation. In any case, we make this assumption here for simplicity and to make the example intuitively clear.

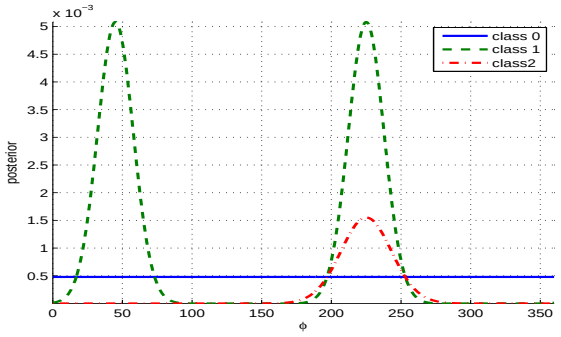



Fig. 5. The posteriors densities after the second measurement.

In the next time step we move to the other side of the object and show how confusion aids in removing the ambiguity.

`Time 3`

The robot position is given by $\mathbf{p} = \mathbf{X}(\phi_1, \mathbf{x}_1) \circ [0\ 10\ 0]^\top$. The measurements are given by $\mathbf{y}_1(3) = [\widehat{c}_1\ \widehat{c}_2]^\top$. The posterior density functions are shown in Figure 6.

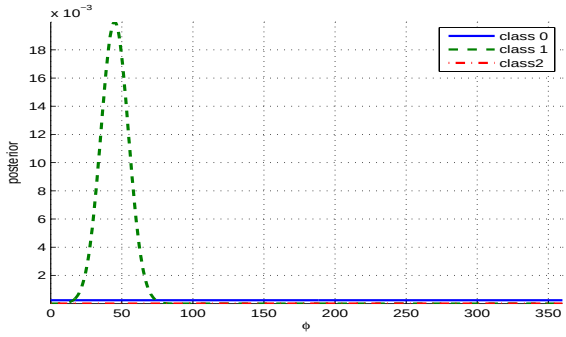



Fig. 6. The posteriors densities after the third measurement.

We compute $\mathrm{P}(c_0|\mathbf{y}_1, o_1) = 0.1502$, $\mathrm{P}(c_1|\mathbf{y}_1, o_1) = 0.8498$ and $\mathrm{P}(c_2|\mathbf{y}_1, o_1) = 1.8 \times 10^{-5}$ at time 3. Then for the maximum class posterior estimate $\widetilde{c}_1$ we compute the maximum $\mathrm{argmax}_{\phi_1}\, p(c_1, \phi_1, \mathbf{x}_1|\mathbf{y}_1(1), o_1)$ which is now unique $\widetilde{\phi}_1 \approx 45^o$. The orientation estimate is non-ambiguous in this case since we exploited inter-class confusion.

Note that we have estimated the orientation quite accurately using only measurements of the object class label and a pre-defined heuristic spatial likelihood function. We believe this is a novel result in the sense of minimalistic sensing[7].

## IV. Grid-Based Object Classification and Orientation Estimation

Consider a grid on $\mathbb{R}^2$ denoted by $\mathcal{G}$. For simplicity, we assume the grid $\mathcal{G}$ consists of $n_g$ grid squares of uniform size (the generalization to nonuniform grid cells is straightforward). Each grid square is denoted by $g_i \in \mathcal{G}$ and can be characterized by the center point $\mathbf{g}_i \in \mathbb{R}^2$. We are interested in assigning to each cell $g_i$ the probability density $p(c_i, \phi_j, \mathbf{x}_j|\mathbf{y}_j, g_j)$ from which we can determine the probability $\mathrm{P}(c_i|\mathbf{y}_j, g_j)$ via marginalization. In fact, for each cell we assign $n_c$ such probability densities - one for each class. Then $\sum_i \mathrm{P}(c_i|\mathbf{y}_j, g_j) = 1$ where $i = \{0, \ldots, n_c\}$ for each cell. In practice a lot of the cells will be dominated by the probability value $\mathrm{P}(c_0|\mathbf{y}_j, g_j)$.

In this scenario, $\mathbf{x}_j$ is the location of the $j$'th cell $g_j$ and is known. If we imagine a robot located at $\mathbf{s}$ with $\vartheta_j$ the direction to $o_j$ defines a ray which we limit to the length $d$. We update the set of cells $\{g_j\}$ that intersect such a ray using the posterior density formula given in the previous sections. The value $\mathbf{x}_j$ is taken as the cell center $\mathbf{g}_j \in \mathbb{R}^2$ and thus cells close or far from the robot (along the ray) are likely to be estimated as $c_0$. We could also define a conic region, e.g. by defining two rays using the bounding box of the object in the image, and then update the cells which intersect the conic region, e.g. see Figure 7.

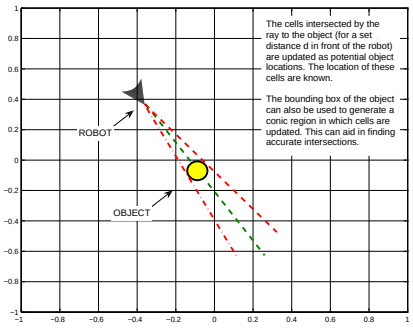



Fig. 7. An example grid environment.

The grid-based estimation problem follows closely the examples given in the last subsection where the location of each cell is known and is analogous to the location of an object position. For simplicity we have assumed cell independence. It is possible to relax this assumption but there are difficulties in doing so that are beyond the scope of this paper. For the grid-based scenario we will examine a practical experiment which is outlined in the next section.

## V. Experimental Results

We conduct our experiments on the EU FP7 project CogX robot platform. The robot is equipped with a Point Grey Flea stereo camera (only one camera used in the experiment) on top of a Pioneer P3X robot base; see Figure 8.

We use FERNS as an object class detector [11], [12]. The robot position is computed from only odometry and the grid organization is known (each grid cell is 2 square decimeters).

The robot is in a room with three objects, $o_1$ is a box containing physics books and $o_2$ and $o_3$ are identical boxes containing robot parts. The boxes are located as shown in Figure 8. All objects have the same CAS lab logo on one of their sides and cannot be differentiated based on the class returns when viewed from this side. We call this *the confusion side* of the object. On the polar opposite side, $o_1$ exhibits a label indicating the box contains physics books whereas both $o_2$ and $o_3$ contain identical labels indicating

[7]The sequence of measurements (and confusions) affect the evolution of the posterior densities in interesting ways but we cannot explore all the cases here. In the experimental section more examples are given.

they contain robot parts. Views of this distinguishing box label are said to be of the *non-confusion side* of the object. The true orientations for $o_1$, $o_2$ and $o_3$ are $\phi_1 = 255°$, $\phi_2 = 315°$ and $\phi_3 = 180°$.

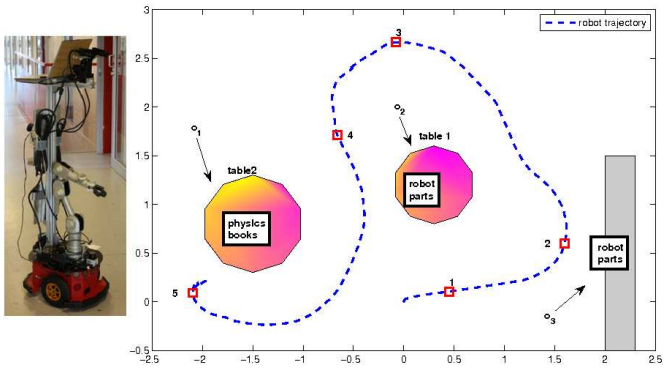



Fig. 8. [Left] CogX robotic platform and [Right] the robot trajectory and layout of the environment.

The robot starts at $(0, 0)$ and follows the trajectory shown in Figure 8. Numbered positions in Figure 8 are where the robot takes a class label measurement. The non-object class, physics books box and robot parts box are labeled as $c_0$, $c_1$ and $c_2$ respectively. In this scenario the units are decimeters. As such, the likelihood functions $p(\widehat{c}_i, o_j | c_i, \phi_j, \mathbf{x}_j)$ for $i \in \{1, 2\}$ and $j \in \{1, 2, 3\}$ are identical to those defined in (25) in the simulated example problem. Similarly, $p(\widehat{c}_i, o_j | c_k, \phi_j, \mathbf{x}_j)$ for $i \neq k \in \{1, 2\}$ are identical to the likelihood functions defined in (26). Finally, $p(\widehat{c}_i, o_1 | c_0, \phi_1, \mathbf{x}_1)$ for $i \in \{1, 2\}$ is identical to the function defined in (27). The recognition system can of course return class measurements $\widehat{c}_1$ and $\widehat{c}_2$.

### A. Orientation Estimation at the Correct Grid Cell

The estimation algorithm in this section is run over a grid as discussed in the last section. However, to visualize the orientation estimate's density we need to essentially look at an individual cell. Thus, in this subsection we examine the orientation estimate in the practical experiment at the true object grid cell. Later we examine the grid map for the environment and show the distribution of the class label probabilities over a number of cells.

At point 1, the robot detects $o_2$ on its confusion side, i.e. both $\widehat{c}_1$ and $\widehat{c}_2$ are measured. The resulting orientation estimates for each class are shown in Figure 10 part (a). Since both $\widehat{c}_1$ and $\widehat{c}_2$ are detected and no further information is available, the probability estimates for both classes are equal but the maximum a posterior orientation estimate is non-ambiguous. The orientation estimate $\widetilde{\phi}_2 \approx 317°$ which is relatively close to the true orientation estimate.

At point 2, the robot detects $o_3$ from a non-confusion side; see Figure 9. The class measurement is only $\widehat{c}_2$ since the observed side is a discriminative one. However notice that the orientation estimate is multi-modal with $\widetilde{\phi}_3 \approx 3°$ and $\widetilde{\phi}_3 \approx 183°$. Since this box is on a shelf against a wall, it is not possible to observe it from other sides. We will show in the following how observing the confusion side improves the overall orientation estimate.

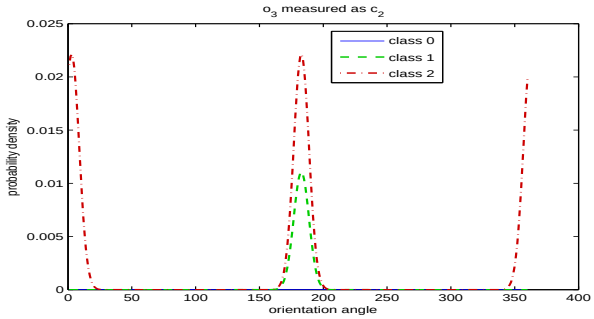



Fig. 9. The robot measures $c_2$ only. Note that the distribution is multi-modal. No further measurements are taken of this object.

At point 3, the robot observes a non-confusion side of $o_2$, i.e. only $\widehat{c}_2$ is measured which is the true class of $o_2$; see Figure 10 part (b). Notice that the probability over $\phi_2$ for the class $c_1$ has dropped and will continue to do so as more measurements of $\widehat{c}_2$ are acquired.

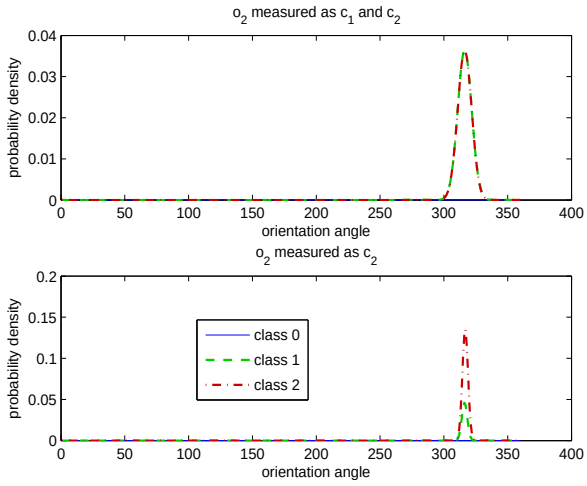



Fig. 10. [Top] The object is first seen from its confusion side. This is not enough to determine the class therefore class estimates are equal. [Bottom] The observation from its non-confusion side helps improving the class estimates.

At point 4, the robot detects the $o_1$ from its non-confusion side, i.e. only $\widehat{c}_1$ is measured which is the true class of $o_1$. As with the first measurement of $o_3$, we have two peaks for the detected class shown in Figure 11. The orientation estimate is given by $\widetilde{\phi}_1 \approx 57°$ and $\widetilde{\phi}_1 \approx 237°$. The maximum class estimate is $\widetilde{c}_1$.

At point 5, the robot observes $o_1$ from its confusion side. In this case since the object has been detected once from its non-confusion side, the probability of $o_1$ being of class $c_1$ is now much higher and the orientation estimate is now non-ambiguous with $\widetilde{\phi}_1 \approx 258°$ as shown in Figure 11. We now see that the confusion side helps to eliminate one of the peaks in the orientation estimate and the spatial likelihood function has helped the estimate converge to an accurate value.

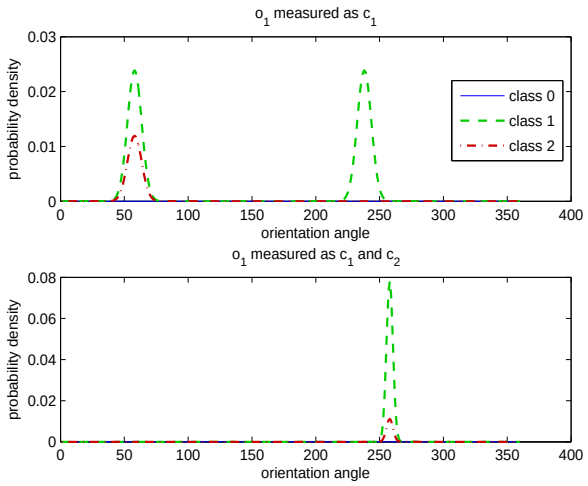
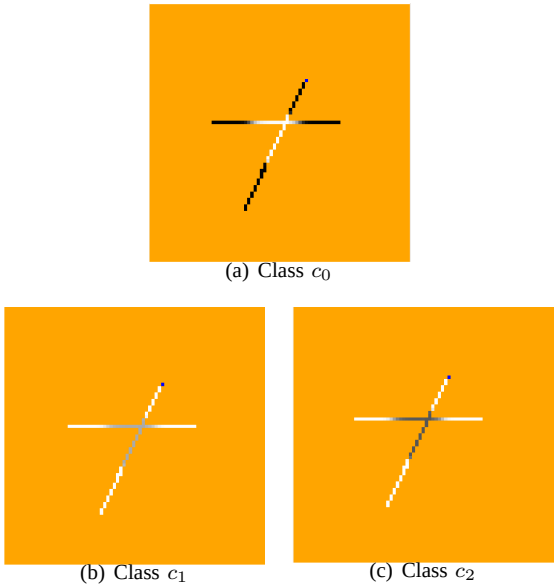



Fig. 11. [Top] The robot first measures $c_1$ and then [Bottom] both $c_1$ and $c_2$. Notice even though the confusion of the second measurement improves the orientation estimation.



(a) Class $c_0$ (b) Class $c_1$ (c) Class $c_2$

Fig. 12. An example distribution over a grid where $o_2$, which is of true class $c_2$, is seen from points 3 and 4 in Figure 8.

### B. Class Probability Estimation over the Grid Map

As described previously, the estimation algorithm is executed over a grid where the detected object rays define a set of cells updated after each measurement. Each cell is associated with an orientation and class density (for all possible classes). As an example, the marginalized probabilities for $c_0$, $c_1$ and $c_2$ are visualized in Figure 12 for class measurements of $o_2$ (recall we have assumed that particular class measurements can be assigned to the correct object).

In this particular snapshot, $o_2$, which is of true class $c_2$, is seen from two positions (points 3 and 4 in Figure 8) and two rays are cast. The gray shading in each picture along the rays represents the probability $\mathrm{P}(c_i, o_2|y_2)$ for each respective class. We have normalized the shading so $\mathrm{P}(c_i, o_2|y_2) = 0$ is pure white while $\mathrm{P}(c_i, o_2|y_2) = 1$ is pure black. The orange background is used here to simplify visualization but can be thought of as the initial prior class probabilities for all cells (i.e. equal priors for all classes) and remains valid since these cells are not updated given only these measurements.

In Figure 12 we can note the probability $\mathrm{P}(c_2, o_2|y_2)$ along the rays increases in magnitude up until the grid cells located at the approximate object location, i.e. the intersection point of the rays. It then decreases as expected. Similarly, $\mathrm{P}(c_o, o_2|y_2)$ decreases in magnitude along the rays until the intersection where it is almost zero and then begins to increase as expected further away from the object.

## VI. Conclusion

We have provided a solution to the problem of simultaneous object class and pose estimation using a generic object classifier and a spatially dependent measurement likelihood model. Our novelty is the ability to estimate both the class and pose of the objects in the environment given only measurements of the object class label from a generic classifier.

We believe the heuristics behind the design of the likelihood functions are realistic. However, one practically and theoretically interesting direction for future work includes the development of reinforcement-like learning algorithms for estimating the likelihood function parameters online. Another interesting direction for future work involves the design of control algorithms for actively searching the environment in order to maximize the information gain.

## References


[1] R. Bajcsy. Active perception. *Proceedings of the IEEE*, 76(8):966–1005, August 1988.

[2] S. Vasudevan, S. Gchteraa, V. Nguyena, and R. Siegwart. Cognitive maps for mobile robots - An object based approach. *Robotics and Autonomous Systems*, 55(5):359–371, May 2007.

[3] P.J. Besl. Geometric modeling and computer vision. *Proceedings of the IEEE*, 76(8):936–958, August 1988.

[4] D.F. Dementhon and L.S. Davis. Model-based object pose in 25 lines of code. *International Journal of Computer Vision*, 15(1-2):123–141, June 1995.

[5] G. Dudek and C. Zhang. Vision-based robot localization without explicit object models. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA'96)*, pages 76–82, April 1996.

[6] M.A. Sipe and D. Casasent. Global feature space neural network for active computer vision. *Neural Computing and Applications*, 7(3):195–215, September 1998.

[7] C.-P. Lu, G.D. Hager, and E. Mjolsness. Fast and globally convergent pose estimation from video images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(6):610–622, June 2000.

[8] H. Murase and S.K. Nayar. Visual learning and recognition of 3-d objects from appearance. *International Journal of Computer Vision*, 14(1):5–24, January 1995.

[9] H. Borotschnig, L. Paletta, M. Prantl, and A. Pinz. Appearance-based active object recognition. *Image and Vision Computing*, 18(9):715–727, June 2000.

[10] B.D.O. Anderson and J.B. Moore. *Optimal Filtering*. Prentice Hall, Englewood Cliffs, N.J., 1979.

[11] M. Ozuysal, P. Fua, and V. Lepetit. Fast keypoint recognition in ten lines of code. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR'07)*, pages 1–8, June 2007.

[12] M. Ozuysal, M. Calonder, V. Lepetit, and P. Fua. Fast keypoint recognition using random ferns. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, accepted, 2009.

# Object search on a mobile robot using relational spatial information

Alper AYDEMIR, [1] Kristoffer SJÖÖ and Patric JENSFELT
*Centre for Autonomous Systems, Royal Institute of Technology, Sweden*

**Abstract.** We present a method for utilising knowledge of qualitative spatial relations between objects in order to facilitate efficient visual search for those objects. A computational model for the relation is used to sample a probability distribution that guides the selection of camera views. Specifically we examine the spatial relation "on", in the sense of physical support, and show its usefulness in search experiments on a real robot. We also experimentally compare different search strategies and verify the efficiency of so-called indirect search.

**Keywords.** Indirect search, Active visual search, Spatial relations, Qualitative spatial reasoning

## Introduction

The ability to find objects in a 3D world is an important item on a mobile robot's skill repertoire. Previous work on object search stems mainly from the field of computer vision. Ideally a robot with a specific task of locating an object should make use of all the bits and pieces of evidence; be it from an overheard dialogue, target object's class limiting the search to a specific region (e.g. forks are usually found in kitchen) or a known spatial relation between the target and some other entity. Some work concentrates on locating the target in the image, thus assuming that the target is already in the field of view [7]. Others investigate algorithms for covering a known or previously unknown world efficiently [1,5,8,9,10].

One powerful idea which naturally involves integration of multiple cues is *indirect search* [3]. Indirect search is about first looking for an intermediate object in order to find the target object by exploiting the relation between the former and the latter. This can be exemplified by first searching for the larger and easier-to-detect whiteboard, and then looking for the pen next to it. To be practical, the system needs to make a decision on which approach to choose based on some criteria. Although this is a simple idea, accomplishing it by fusing multiple types of cues can prove to be hard and is not yet in place in the previous work.

The novelty of this paper is given by an investigation of the following question: Is it possible to make use of spatial relations in order to aid a mobile robot tasked with finding an object? For this particular work we have chosen to investigate the relation of

[1]Corresponding Author: Alper Aydemir, Royal Institute of Technology (KTH), Centre for Autonomous Systems, SE-100 44 Stockholm, Sweden; E-mail: aydemir@csc.kth.se

physical support, i.e. *on*. We introduce a computational perceptual model for the physical support relation, and show how algorithms using this model can significantly increase the efficiency of visual object search, illustrating the fact through real world experiments. In this way, we believe that the work presented here takes a more principled approach towards indirect search compared to previous work.

## 1. Spatial Relations as functions

Spatial relations between entities are important in human cognition, as evidenced by the prolific use of spatial prepositions in language, in both concrete and metaphorical contexts. Here, we are interested in using the information carried by a relation between two objects $A$ and $B$, together with the location of one of them, for the purpose of locating the other efficiently.

We regard a spatial relation as a function, dependent on the objects involved, from the space of all the objects' possible poses, to the interval $[0, 1]$:

$$\mathcal{R}_{A,B} : \{\pi_A, \pi_B\} \rightarrow [0, 1] \tag{1}$$

where $1$ represents that the relation is completely fulfilled by the pose combination, and $0$ that the relation does not apply at all. The resulting value, despite being in the range $[0, 1]$, is not a probability. However, it is possible to obtain a probability distribution over poses implicitly from this function, as shown below.

### *1.1. ON*

As a good example of a spatial relation that will be useful for a robot in a search scenario, we have chosen "on". "On" is one of the most fundamental prepositions in the English language, and represents a highly relevant functional relationship between many objects in our environment [4,6]; thus, a robot will often have information about an object's location in terms of it being "on" something else – this information could come from dialogue with humans, from commonsense rules for the typical behaviour of objects, or from a statistical model learned from experience over time.

The central functional aspect of the word "on" is the *support* that one object gives another. Humans learn to judge this with experience, manipulating and observing; for now, robots must rely on short-cuts. We therefore propose a perceptual geometric model intended to estimate how well the relation between two objects corresponds to one of support. The model is defined using the following criteria ($O$ denotes the *trajector* object, i.e. the object that is "on" the other, and $S$ the support object or *landmark*). The proposed function is termed $\textsc{On}(O, S)$. The criteria are illustrated in Figure 1; they are:

- *Separation between objects*, $d$. $d$ can be positive or negative, negative values meaning that objects are, or seem to be, interpenetrating.
  In order for an object to mechanically support another, they must be in contact. Due to imperfect visual input and other errors, however, contact may be difficult to ascertain precisely. Hence, to create a soft constraint, the apparent separation is used as a penalty.

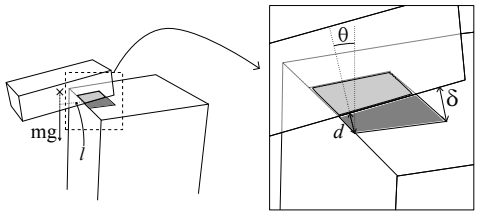



**Figure 1.** Key features used in computation of ON: Separation $d$, COM offset $l$, contact angle $\theta$ and contact threshold $\delta$. The gray area represents the contact.

- *Horizontal distance between COM and contact, $l$.* It is well known that a body $O$ is statically stable if its center of mass (COM) is above its area of contact with another object $S$; the latter object can then take up the full weight of the former. Thus we impose a penalty on $\text{ON}(O, S)$ that increases with the horizontal distance from the contact area to the COM of $O$. The contact area is taken to be that portion of $S$'s surface that is within a threshold, $\delta$, of $O$, in order to deal with the uncertainties described above. If $d > \delta$, the point on $S$ closest to $O$ is used instead; otherwise, $l$ is the positive distance to the outer edge of the contact area if outside it, and the negative distance if inside.
- *Inclination of normal force, $\theta$* – the angle between the normal of the contact between $O$ and $S$ on the one hand, and the vertical axis on the other. The reason for including this is that, all other things being equal, the normal force decreases as the cosine of $\theta$, meaning the weight of $O$ must be either supported by another object or by friction (or adhesion).

All these values can be computed from visual perception in principle. The position of the COM is taken as the average point of the objects' geometry (since density cannot be determined by vision), unless otherwise known in advance.

The first criterion is evaluated as the *distance factor* in an exponential function:

$$\text{ON}_{distance}(O, S) \triangleq \exp\left(-\frac{d}{d_0(d)} \ln 2\right) \tag{2}$$

where $d_0$ is the falloff distance at which ON drops by half:

$$d_0 = \begin{cases} -d_0^-, d < 0 \\ d_0^+, d >= 0 \end{cases}$$

The constants $d_0^-$ and $d_0^+$ are both greater than $0$ and can have different values (representing the penalty for the penetrating and nonpenetrating case, respecively).

The latter two criteria make up the *contact factor*:

$$\text{ON}_{contact}(O, S) \triangleq \sin\theta \cdot \frac{1 + \exp(-(1-b))}{1 + \exp\left(-\left(\frac{-l}{l_{\max}} - b\right)\right)} \tag{3}$$

Here, $l_{\max}$ is the maximum possible distance an internal point can have within the contact area, and $b$ is an offset parameter.

The exact expressions for the factors (2) and (3) are not central here; what matters is that they yield the applicability 1 for the ideal case for each criterion, and drop off to 0 as the criterion is violated, while being "soft" in order to be robust to error.

The values are combined by choosing whichever factor is smaller, indicating the greater violation of the conditions for support:

$$\textsc{On}(O, S) \triangleq \min(\textsc{On}_{contact}, \textsc{On}_{distance}) \tag{4}$$

### *1.2. Probability modelling*

Although the conceptualization above does not explicitly make use of any probabilities, it is obvious that the fact of an object being ON another is not sufficient to recover the exact pose of the trajector. A probability distribution over poses can be produced in the following way:

Given the pose and geometry of the landmark $S$, and the geometry (but not the pose) of the trajector $O$, each possible pose $\pi$ for the trajector yields a value of $\textsc{On}(O_\pi, S)$ for that pose.

It is now possible to introduce probabilities in the following way. Introduce a true/false event $On(O, S)$ signifying that $\textsc{On}(O, S) > t$ where $t$ is a threshold. Then,

$$p(\pi|On(O_\pi, S)) = \frac{p(On(O_\pi, S)|\pi)p(\pi)}{p(On(O_\pi, S))} = \tag{5}$$
$$= \frac{[\textsc{On}(O_\pi, S) > t]p(\pi)}{p(On(O_\pi, S))}$$

Here [] denotes the Iverson bracket:

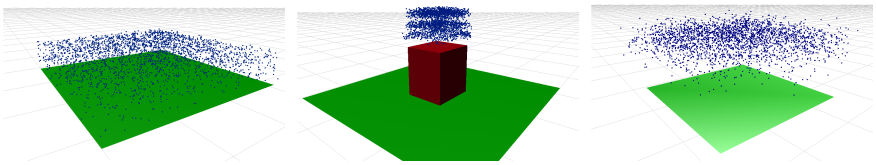
$$[X] = \begin{cases} 1, \text{ if } X \text{ is } \textsc{True} \\ 0, \text{ otherwise} \end{cases}$$

In other words, the probability is simply proportional to the prior for the pose $\pi$ whenever $\textsc{On}(O_\pi, S) > t$, and 0 elsewhere. Though it may be hard to express this distribution analytically, by drawing samples randomly from $p(\pi)$, discarding those failing to reach the threshold, and normalising over the remainder, an arbitrarily good approximation can be found. In the following, we use a $t$ value of 0.5.

Figure 2 shows simulated examples of distributions sampled according to the above. 2(c) shows *chained* sampling: an object is ON another, which is ON the table, but both have unknown poses. First the bottom object is sampled, and for each sample that passes the threshold, the top object is sampled in turn. The uncertainties of both objects add up, resulting in a more diffuse point cloud at a greater height above the table.

## 2. Object Search

The goal of the object search process performed by a mobile robot is to calculate a set of sensing actions with minimum cost which brings the target object, in whole or partly, into the sensor field of view so as to maximize the target object detection probability.



(a) Object on table (b) Object on box (c) Object on box on table

**Figure 2.** Simulated examples of sampled distributions of ON

Here we briefly give a formulation of the object search problem using the notation of [10]. Let $\Psi$ be the $2D$ search region whose structure is known *a priori*. To discretize the search region, $\Psi$ is tessellated into identically sized cells, $c_1 ... c_n$. The area outside of the search region is represented by a single cell $c_0$. A sensing action $s$ is then defined as taking an image of $\Psi$ from a view point $v$ and running a recognition algorithm to determine whether the target object $o$ is present or not. In the general case, the parameter set of $s$ consists of camera position $(x_c, y_c, z_c)$, pan-tilt angles $(p, t)$, focal length $f$ and a recognition algorithm $a$; $s = s(x_c, y_c, z_c, p, t, a)$. The cost of a search plan $S = s_0 ... s_i$ is then given as $C(S)$.

A search agent starts with an initial probability distribution (PDF) on target object location over $\Psi$. We assume that there is exactly one target object in the environment either inside or outside the search region. This means that all cells will be dependent and every sensing action will influence the values of all cells. Let $\beta$ be a successful detection event and $\alpha_i$ the event that the center of $o$ is at $c_i$. The probability update rule after each $s$ with a non-detection result is then:

$$\mathbf{p}(\alpha_i|\neg\beta) = \frac{\mathbf{p}(\alpha_i)(1 - \mathbf{p}(\beta|\alpha_i))}{\mathbf{p}(\alpha_0) + \sum_{j=1}^{n} \mathbf{p}(\alpha_j)(1 - \mathbf{p}(\beta|\alpha_j))} \tag{6}$$

Note that for $i = 0$ , $\mathbf{p}(\beta|\alpha_i) = 0$, i.e. we cannot make a successful detection if the object is outside the search region . Therefore after each sensing action with a non-detection result the probability mass inside $\Psi$ shifts towards $c_0$ and the rest of $\Psi$ which was not in field of view.

### *2.1. Next best view selection*

The next step is to define how to select the best next view given a PDF. First, candidate robot positions are generated by randomly picking samples from the traversable portion of $\Psi$. This results in several candidate robot poses each with associated view cones. For a given camera, the length of the view cone is given by the greatest distance at which the object can reliably be detected, which depends on the size of the object.

The next best view point is then defined as:

$$\underset{j=1..N}{\operatorname{argmax}} \sum_{i=1}^{n} \mathbf{p}(\alpha_i) V(c_i, j) \tag{7}$$

Where $N$ is the number of candidate view points and $V$ is defined as:

$$V = \begin{cases} 1, \text{ if } c_i \text{ is inside of the } j^{\text{th}} \text{ view cone} \\ 0, \text{ otherwise} \end{cases}$$

## 3. Experiments

### *3.1. Implementation Details*

The robot used in our experiments is a Pioneer III wheeled robot, equipped with a Hokuyo URG laser range finder and a stereo camera (with no zoom capability) mounted on a pan-tilt unit at 1.4 m above the ground. The system uses a SLAM implementation [2] for localization and mapping and builds an occupancy gridmap based on laser data. The experiments were carried out in a mock-up living room (Figure 3). Two planar objects – a low table and a large desk – were present in the experimental area, and their poses known to the system. The detectable objects used were a large cardboard box and small rice carton (see Figure 4). Preparatory experiments showed that the threshold distance, at which the objects were detected at least 75% of the time, was 1 m and 4 m for the small and the large object, respectively. These were the maximum distances used in the view cone generation (see Section 2.1).

**Figure 3.** Experimental environment and robot platform

During experiments, the larger box and the rice carton were placed randomly on one of the tables, at a 50% chance for each. In order to minimize the bias, different people from our lab, unconnected with the research, were asked to "put the box on the table/desk and rice carton on the box". The objects were free to be placed in any orientation and pose provided they are placed on their physical support object.

In order to assign a prior to the grid cells (Section 2) we generated random samples as described in Section 1.2 and used KDE. 150 samples that passed the threshold $\text{ON} > 0.5$ were convoluted with a simple 2D Epanechnikov kernel:

$$K(u) = c \cdot (1 - u^2)$$

with a kernel radius chosen to be 0.2 m. The resulting grid was then normalized.


**Figure 4.** Test objects: “rice” and “printer”

The object search was carried out as described in Section 2. The initial information given to the system was:

1. The *a priori* probability that the object sought was in fact in the room was given at 80% (i.e. $\mathbf{p}(c_0) = 0.2$).
2. The “rice” object was ON the “printer” object with 100% certainty.
3. The “printer” object was ON either the table or the desk, each with 50% probability.

When the best next view was decided on, the robot moved to the corresponding position and orientation. 25 pose samples for the target object (with ON above the threshold 0.5) were then obtained from the region of the view cone, and their average used to set the tilt angle of the camera in order to capture the most likely object height.

Object detection and pose estimation was done using previously trained SIFT features. The generation and processing of new views was kept up until either the “rice” object was found, or until the search was considered to have failed. The criterion for failure was a posterior probability of 70% that the object was not inside the room. We performed three types of searches utilising the prior information to varying degrees; uninformed search, chained inference with 2 relations and indirect search with 2 relations. In the following we will denote the rice carton by *A* and the cardboard box by *B*.

### *3.2. Chained inference with 2 relations*

In this test, the information given was that $B$ was ON a table, and that $A$ was ON $B$, but otherwise $A$ and $B$ had unknown poses. The robot is tasked to look directly for $A$. By making use of the a priori information via chained inference, as described in Section 1.2, a probability distribution was sampled for $A$’s pose, and visual search was planned using this distribution directly. Figure 5 shows the robot processing a view during this search. Note the tilt of the camera, illustrating the robot’s expectation for the vertical position of the object, given that it is supposed to be on top of the larger box.

### *3.3. Indirect search with 2 relations*

In this scenario, the robot exploits the position of the relatively more easily detectable $B$ to find $A$. The initial information provided to the robot is the same as in Section 3.2. However, this time the system first sampled the distribution of $B$ (given that it was on a table) and performed the visual search for $B$ based on the resulting probability distribu-

**Figure 5.** Chained inference, direct search: While searching for the rice carton, the robot looks towards the height of the target object had it been on top of the large box object.


(a) (b)

**Figure 6.** (a) The robot first finds the cardboard box which can be detected easily as opposed to rice carton. (b) Once the cardboard box is found, the search space is greatly reduced and the rice carton is found with the next view.

tion. Only if and when $B$ was found did the system compute the distribution of $A$ using this new data, using that distribution in its turn to perform a focused search for $A$. Note that by finding $B$ and generating possible poses for $A$ the robot reduces the search space significantly. The experimental results also bear this out. Figure 6 shows the robot as it detects the larger box at a distance, then closes to locate the "rice" object at a distance where the model indicates that detection is likely.

### *3.4. Uninformed search*

As a baseline, we ran the algorithm without utilising the information in the spatial relation. Thus, item 3 in the above list of *a priori* knowledge was not used. Instead, the visual search for $B$ used a prior PDF that simply assigned a uniform probability for the object to all obstacles registered by the laser scanner. In lieu of the vertical information otherwise provided by the spatial relation, the camera instead tilted to a set sequence of: down 30°, straight forward, and up 30°. When the $B$ object was detected, the conditioned probabil-

| Mode | % success | Avg. # views, failure | Avg. # views, success |
|---|---|---|---|
| Direct chained | 73 | 5 | 5 |
| Indirect chained | 93 | 5 | 2 |
| Uniform | 46 | 17 | 10 |

**Figure 7.** Results of experimental evaluation

ity for $A$ was used as in 3.3. The reason for not conducting an uninformed search directly for $A$ over the whole space is that this proved infeasible in experiments, as the number of view points invariably exceeded our limit of $20$. The fail search criterion was also not met because the smaller view cones resulting from the object's smaller size shifted little of the probability mass out towards $\mathbf{p}(c_0)$, the probability that the target object is outside of the search space. This is in contrast to the larger "cardboard" object where after each non-detection a substantial amount of probability mass flowed towards $\mathbf{p}(c_0)$.

### *3.5. Results*

For each of 15 different object configuration, all three types of searches were performed for a total of 45 runs. We present the results of our experiments in Table 7.

By comparing uniform and direct search, the advantage of using the spatial relation knowledge is evident. Ignoring the information that the printer box is on the table leads to unnecessary views of the walls and other irrelevant obstacles. Also the lack of vertical position information necessitates redundant image processing as the camera goes through 3 tilt angle settings in order to ensure vertical coverage.

The difference in performance between indirect and direct search illustrates the usefulness of indirect search, even when the spatial relations are taken into account fully. Chained sampling allows the robot to directly create a probability representation for the sought object, bypassing the search for the larger object and providing an approximate height at which to aim the camera; nevertheless, the small size of the object means that many views may be necessary to cover a large area . However in the indirect search case, once the larger object is located then the search space is greatly reduced and typically the target object is found within the next view or two.

## 4. Conclusions

We have proposed a way in which spatial relations, in the form of applicability functions, can be used to aid in visual object search. We suggested a perceptual geometrical model that approximates the core meaning of the topological preposition "on", i.e. the notion of support. In experiments on real robot, running autonomously, we have shown the advantages to being able to incorporate information about support into a visual search framework:

- Knowing that a relation holds between an object of known pose and one of unknown pose allows for limiting the 2D space over which to search for the latter.
- Indirect search can help with the localization of a smaller object, by allowing the search to start with a larger, easier-to-detect object.
- The support property can be used to guide the search in the vertical dimension.

The results reinforce the notion that indirect search is a useful method in active visual search; our contribution here is the expression of how indirect search is done in conjunction with qualitative spatial relations, as well as the specific instantiation using the ON relation.

## 5. Discussion

In this work, experiments were relatively limited in scope and served only to compare different search modes with each other. One avenue of investigation is to vary the parameters of the objects involved; for example, changing the characteristics of the involved objects to find the threshold where indirect search becomes more costly than chained search.

The inclusion of other qualitative relations is another interesting direction for further research; especially other topological relations such as “in”, “near”, and “at”, as these are all to some extent objective and functional in nature.

The search problem formulation used herein is also rather simplistic, counting the cost of a search merely in the number of views processed. The formulation also presupposes a “one-shot” visual system, as opposed to a continual one. The visual search algorithm would necessarily change under a different problem formulation; however, the way spatial relations are included in the solution need not be much changed, we believe.

## Acknowledgements

The authors are with the Centre for Autonomous Systems at the Royal Institute of Technology (KTH), Stockholm, Sweden. This work was supported by the SSF through its Centre for Autonomous Systems (CAS), and by the EU FP7 project CogX. K. Sjöö was additionally funded by the Swedish Research Council, contract 621-2006-4520

## References

[1] S. Ekvall, D. Kragic, and P. Jensfelt. Object detection and mapping for service robot tasks. *Robotica: International Journal of Information, Education and Research in Robotics and Artificial Intelligence*, 2007.

[2] J. Folkesson, P. Jensfelt, and H. Christensen. The m-space feature representation for slam. *IEEE Transactions on Robotics*, 23(5):1024–1035, Oct. 2007.

[3] T. D. Garvey. *Perceptual strategies for purposive vision*. PhD thesis, Stanford, CA, USA, 1976.

[4] A. Herskovits. *Language and Spatial Cognition*. Cambridge University Press, 1986.

[5] K. Sjö, D. Gálvez López, C. Paul, P. Jensfelt, and D. Kragic. Object search and localization for and indoor mobile robot. *Journal of Computing and Information Technology*, 17(1):67–80, March 2009.

[6] L. Talmy. Force dynamics in language and cognition. *Cognitive Science*, 1988.

[7] A. Torralba, M. S. Castelhano, A. Oliva, and J. M. Henderson. Contextual guidance of eye movements and attention in real-world scenes: the role of global features in object search. *Psychological Review*, 113:2006, 2006.

[8] J. K. Tsotsos and K. Shubina. Attention and visual search : Active robotic vision systems that search. In *International Conference on Computer Vision Systems ICVS'07*, Washington, DC, USA, 2007.

[9] L. E. Wixson and D. H. Ballard. Using intermediate objects to improve the efficiency of visual search. *Int. J. Comput. Vision*, 12(2-3):209–230, 1994.

[10] Y. Ye. *Sensor planning for object search*. PhD thesis, 1998.

# Stochastically Convergent Localization of Objects by Mobile Sensors and Actively Controllable Relative Sensor-Object Pose

Adrian N. Bishop and Patric Jensfelt

***Abstract*— The problem of object (network) localization using a mobile sensor is examined in this paper. Specifically, we consider a set of stationary *objects* located in the plane and a single mobile nonholonomic *sensor* tasked at estimating their relative position from range and bearing measurements. We derive a coordinate transform and a relative sensor-object motion model that leads to a novel problem formulation where the measurements are linear in the object positions. We then apply an extended Kalman filter-like algorithm to the estimation problem. Using stochastic calculus we provide an analysis of the convergence properties of the filter. We then illustrate that it is possible to steer the mobile sensor to achieve a relative sensor-object pose using a continuous control law. This last fact is significant since we circumvent Brockett's theorem and control the relative sensor-source pose using a simple controller.**

## I. Introduction

This paper considers the problem of *object* localization using a mobile *sensor* taking relative range and bearing measurements [1], [2]. Furthermore, we consider the problem of actively steering the mobile sensor to achieve a desired relative sensor-object pose with respect to individual objects. The term *object* can be interpreted loosely and might refer to a transmitting node or a target such as an aircraft or missile etc. Alternatively, an object might refer to an everyday object of interest that is to be manipulated by a mobile autonomous robot in an industrial or home environment.

The idea behind the second (control) problem considered in this paper is that it is often the case that a sensor can better localize a target from a particular position or given a certain relative trajectory [3], [4]. Alternatively, we are interested in the problem of localizing a field of objects which the mobile sensor might then wish to return to and manipulate or analyze from certain relative positions.

For example, consider a robot exploring an unknown environment and tasked at localizing a specified class of objects relative to its current position. Following a period of localization, the robot might be asked to return to a particular object and take visual pictures of the object from certain relative positions; i.e. from a certain distance with a certain relative viewing angle [5]. Alternatively, the robot may be required to return to a particular object in order to manipulate or grasp the object for analysis [6] and this task requires a specified relative robot-object pose. It is these sort of scenarios which motivate the formulations and algorithms considered in this paper.

A.N. Bishop and P. Jensfelt are with the Centre for Autonomous Systems, KTH, Stockholm, Sweden. This work was supported by the Centre for Autonomous Systems (CAS) and the EU FP7 project "CogX".

### A. Contributions of this Paper

The contributions of this paper are related to the polar-like nature of the derived problem formulation and the rigorous convergence analysis provided. Firstly, we introduce the relative sensor-object dynamic model (given a unicycle robot) in polar coordinates which leads to a linear measurement equation. We then outline an extended Kalman filter (EKF) algorithm that can be used to estimate the relative object positions. We rigorously analyze the convergence of the filter and illustrate a condition which will guarantee the mean-square error exponential convergences to a bounded steady state value. The problem formulation for object localization introduced in this paper is a very natural representation which leads to improved estimation performance.

Following the filter analysis we outline the problem of actively steering the unicycle robot to achieve a desired (relative in this case) pose with respect to an object of interest. We derive a simple continuous control law for the sensor's translational and angular velocities that will steer it to a desired relative distance and angle with respect to the object (or an estimate of the object) position. The polar problem formulation advocated in this paper provides a natural representation of this control problem and simplifies the controller design. To achieve a desired unicycle sensor pose in a global Cartesian framework is non-trivial. Moreover, the relative object-sensor distance and the relative object sensor angle is a more natural representation of the desired (intuitive) control objective.

Thus, the overall concept of object localization and active sensor-object pose control (for localization and viewing the object) is naturally derived in a simple way in this paper. This contribution is significant and aims to highlight the benefits of seeking alternative coordinate systems as a means of simplifying certain nonlinear problems in robotics, localization and multi-agent systems control.

## II. Preliminaries

Consider a mobile sensor with a state description $\mathbf{s} = [x\ y\ \phi]^{\mathrm{T}} \in \{\mathbb{R}^2 \times \mathbb{SO}(1,\mathbb{R})\}$. Here $x$ and $y$ are the sensor's Cartesian position coordinates and $\phi$ is the sensor's heading. The sensor dynamics are based on the *unicycle* model,

$$\begin{aligned} \dot{x}_r &= v_r \cos\phi_r \\ \dot{y}_r &= v_r \sin\phi_r \\ \dot{\phi}_r &= w_r \end{aligned} \tag{1}$$

where $v$ is the translational velocity and $w$ is the sensor's angular velocity. Note that there are three state variables in

$\mathbb{R}^2 \times \mathbb{SO}(1,\mathbb{R})$ and only two control inputs. The nonholonomic constraint on the sensor is given by

$$\dot{x}_r \sin\phi_r = \dot{y}_r \cos\phi_r \tag{2}$$

and implies via Brockett's theorem that a desired robot pose $\mathbf{s}^* = [x^* \ y^* \ \phi^*]^{\mathrm{T}}$ can not be asymptotically stabilized using a linear smooth time-invariant control law. We assume the control inputs $v$ and $w$ are known precisely (although we can relax this assumption, we do not do so in this paper).

The environment is populated with a set $\mathcal{V}$ of *objects* or target nodes with $|\mathcal{V}| = n$. Here objects might mean source nodes (e.g. active enemy radars, acoustic sources etc), landmarks or feature points as discussed in the simultaneous localization and mapping literature, or targets such as aircraft, missiles etc. Alternatively, objects might mean everyday objects of interest that are to be manipulated by a mobile autonomous robot in a home/industrial environment.

The Cartesian position of the $i^{th}$ object is denoted by $\mathbf{p}_i = [x_i \ y_i]^{\mathrm{T}} \in \mathbb{R}^2$. The objects are stationary in this case and represent the *map* of the environment which is to be estimated by the mobile sensor. At some time $t$ the sensor can sense a subset $\mathcal{G}(t) \subseteq \mathcal{V}$ of landmarks. At time $t$ the true measurements of object $i$ are given by

$$\begin{aligned} d_i &= \sqrt{(x_i - x)^2 + (y_i - y)^2} \\ \vartheta_i &= \theta_i - \phi = \arctan\left(\frac{y_i - y}{x_i - x}\right) - \phi \\ &\quad \forall i \in \mathcal{G}(t) \end{aligned} \tag{3}$$

where $\vartheta_i = \theta_i - \phi$ is the relative bearing to the $i^{th}$ object in the sensor's internal Cartesian coordinate system, i.e. the Cartesian coordinate system rotated by the sensor's heading. Let $\mathbf{z} = [\mathbf{s}_r \ \mathbf{p}_1 \ \ldots \ \mathbf{p}_n]^{\mathrm{T}}$. The measurements are typically corrupted by a noise process $\mathbf{n}(t)$ and thus we can obtain the measurement equation

$$d\mathbf{y}(t) \triangleq \psi(t)dt = h(\mathbf{z})dt + \mathbf{E}(t)d\mathbf{n}(t) \tag{4}$$

in continuous-time. Here, $\mathbf{n}(t)$ is a zero-mean Weiner process and $\mathbf{E}(t)$ is a measurement noise weighting matrix that can be dependent on the true state. For example, it might be true that the noise present in the range measurements is a fraction of the true range. The measurements and robot dynamics are nonlinear in the chosen coordinate system.

## III. Localization of Objects in Polar Coordinates

One contribution of this paper is a novel localization analysis that takes advantage of the polar-like nature of the relative range and bearing measurements. There is a long history in the bearing-only target tracking literature [1], [7] of working in variants of polar coordinate systems. Here, we derive a relative sensor-object motion model and then formulate an estimation problem that involves linear measurements (which can significantly improve the performance of the EKF as noted in many different example problems [1], [7]).

Recall the true measurements taken by the mobile sensor are given by

$$\begin{aligned} d_i &= \sqrt{(x_i - x)^2 + (y_i - y)^2} \\ \vartheta_i &= \theta_i - \phi = \arctan\left(\frac{y_i - y}{x_i - x}\right) - \phi \\ &\quad \forall i \in \mathcal{G}(t) \end{aligned} \tag{5}$$

where the state $\mathbf{s} = [x \ y \ \phi]^{\mathrm{T}}$ of the sensor and the position of the objects $\mathbf{p}_i = [x_i \ y_i]^{\mathrm{T}} \in \mathbb{R}^2$ are in some external (non-relative) coordinate system. The measurements are nonlinear in the first two components of $\mathbf{s}$ and in $\mathbf{p}_i$, $\forall i$.

Now define the following state variable $\mathbf{r}_i = [d_i \ \vartheta_i]^{\mathrm{T}}$ with $d_i \in (0, \infty)$ and $\vartheta_i \in [-\pi, \pi)$. We will always assume that $d_i \neq 0$ for both theoretical and very practical reasons. The augmented state variable in this section is given by $\mathbf{z} = [\mathbf{r}_1 \ \ldots \ \mathbf{r}_n]^{\mathrm{T}}$ and encompasses the relative sensor-object position for all objects in the set. In practice the state $\mathbf{z}$ can be augmented online when each new object is sensed.

The measurements (5) are linear in $\mathbf{r}_i$ or more generally in $\mathbf{z} = [\mathbf{r}_1 \ \ldots \ \mathbf{r}_n]^{\mathrm{T}}$ and are given by the continuous-time measurement equation

$$d\mathbf{y}(t) \triangleq \psi(t)dt = \mathbf{H}(\mathcal{G}(t))\mathbf{z}dt + \mathbf{E}(t)d\mathbf{n}(t) \tag{6}$$

where $\mathbf{E}(t)$ is not required to be independent of $\mathbf{z}$ (as discussed previously). Here $\mathbf{H}(\mathcal{G}(t))$ is a time-varying linear matrix which is dependent only on the set $\mathcal{G}(t)$ of currently sensed landmarks. For example, if all of the landmarks are sensed and the state variable $\mathbf{z}$ is ordered appropriately, then $\mathbf{H}$ would be the identity matrix.

Consider again a robot that obeys the unicycle model (1) in $\mathbb{R}^2 \times \mathbb{SO}(1,\mathbb{R})$. Then we can write down the following differential equation for the dynamics of $\mathbf{r}_i$,

$$\begin{aligned} \dot{d}_i &= -v\cos\vartheta_i \\ \dot{\vartheta}_i &= \frac{v}{d_i}\sin\vartheta_i - w \end{aligned} \tag{7}$$

which is nonlinear in $\mathbf{r}_i$. Note also that $d_i$ must be bounded away from zero here for technical reasons (although practically this is also logical). Again we assume $v$ and $w$ are known precisely.

### A. On the Observability of the Polar Localization Problem and the Convergence of the EKF-Based Algorithm

In this subsection we will examine and prove a number of results regarding the convergence of an EKF-like algorithm for estimating the relative object state variable.

*1) Error Free Measurements and Dynamics:* We consider first the observability properties of the state $\mathbf{z} = [\mathbf{r}_1 \ \ldots \ \mathbf{r}_n]^{\mathrm{T}}$ with $\mathbf{r}_i = [d_i \ \vartheta_i]^{\mathrm{T}}$ evolving according to (7). We also assume error free measurements of the form $\psi(t) = \mathbf{H}(\mathcal{G}(t))\mathbf{z}(t)$.

*Corollary 1: Assume the robot-landmark dynamics and the measurements are deterministic and error free. The state $\mathbf{r}_i(s) = [d_i(s) \ \vartheta_i(s)]^{\mathrm{T}}$ for some $i \in \mathcal{V}$ and for $s \geq \tau$ or $s < \tau$ can be calculated at any time $t \geq \tau$ if and only if $\mathcal{G}(\tau) \cap \mathbf{r}_i(\tau) \neq \emptyset$ for some instant $\tau$.*

The fact that Corollary 1 is true is not surprising but is provided for completeness.

*2) Error Free Dynamics and Noisy Measurements:* A natural extension to the above result concerns the behavior of an estimate $\widehat{\mathbf{z}}$ of $\mathbf{z}$ when the dynamics of the state $\mathbf{r}_i = [d_i \ \vartheta_i]^{\mathrm{T}}$ are error free and deterministic but the measurements

$$d\mathbf{y}(t) \quad = \quad \mathbf{H}(\mathcal{G}(t))\mathbf{z}dt + \mathbf{E}(t)d\mathbf{n}(t) \tag{8}$$

are corrupted by an additive Weiner process. Naturally, the behavior of any state estimate $\widehat{\mathbf{z}}$ depends on the particular estimator and thus let us consider an estimator of the form

$$d\widehat{\mathbf{z}} = f(\widehat{\mathbf{z}}, v, w)dt + \mathbf{K}(t)\left(d\mathbf{y}(t) - \mathbf{H}(\mathcal{G}(t))\widehat{\mathbf{z}}dt\right) \tag{9}$$

where the function $f_i(\cdot)$ that captures the dynamics of the subspace $\mathbf{r}_i = [d_i \ \vartheta_i]^{\mathrm{T}}$ is given by

$$f_i(\widehat{\mathbf{z}}, v, w) = \begin{bmatrix} -v\cos\widehat{\vartheta}_i \\ \frac{v}{\widehat{d}_i}\sin\widehat{\vartheta}_i - w \end{bmatrix} \tag{10}$$

where $v$ and $w$ are again considered as deterministic control inputs with no errors. The function $f(\cdot)$ is thus a vertical concatenation of the $f_i(\cdot)$. The gain $\mathbf{K}(t)$ is given by

$$\mathbf{K}(t) = \mathbf{P}(t)\mathbf{H}^{\mathrm{T}}(\mathcal{G}(t))\mathbf{R}^{-1}(t) \tag{11}$$

and $\mathbf{P}(t)$ is the solution to the following Riccati differential equation

$$\begin{aligned} d\mathbf{P}(t) \quad = \quad & \left[\mathbf{A}(t)\mathbf{P}(t) + \mathbf{P}(t)\mathbf{A}^{\mathrm{T}}(t) + \mathbf{Q}(t)\right]dt - \\ & \mathbf{P}(t)\mathbf{H}^{\mathrm{T}}(\mathcal{G}(t))\mathbf{R}^{-1}(t)\mathbf{H}(\mathcal{G}(t))\mathbf{P}(t)dt \end{aligned} \tag{12}$$

where $\mathbf{Q}$ and $\mathbf{R}$ are positive-definite tuning matrices. Note that $\mathbf{A}(t)$ is the Jacobian of $f(\cdot)$ evaluated at $\widehat{\mathbf{z}}$. The Jacobian $\mathbf{A}_i(t)$ of $f_i(\cdot)$ is given by

$$\mathbf{A}_i(t) = \begin{bmatrix} 0 & v\sin\widehat{\vartheta}_i \\ -\frac{v}{\widehat{d}_i^2}\sin\widehat{\vartheta}_i & \frac{v_r}{\widehat{d}_i}\cos\widehat{\vartheta}_i \end{bmatrix} \tag{13}$$

and is evaluated at $\widehat{\mathbf{r}}_i$ and is dependent on $v$. Note the estimation error $\boldsymbol{\zeta} = \mathbf{z} - \widehat{\mathbf{z}}$ evolves according to

$$\begin{aligned} d\boldsymbol{\zeta} \quad = \quad & \left(\mathbf{A}(t) - \mathbf{K}(t)\mathbf{H}(\mathcal{G}(t))\right)\boldsymbol{\zeta}dt + \\ & \varrho(\mathbf{z}, \widehat{\mathbf{z}}, v, w)dt - \mathbf{K}(t)\mathbf{N}(t)d\mathbf{n}(t) \end{aligned} \tag{14}$$

where we have used the following Taylor expansion of $f(\cdot)$ about the estimate $\widehat{\mathbf{z}}$,

$$f(\mathbf{z}, v, w) - f(\widehat{\mathbf{z}}, v, w) = \mathbf{A}(t)\left(\mathbf{z} - \widehat{\mathbf{z}}\right) + \varrho(\mathbf{z}, \widehat{\mathbf{z}}, v, w) \tag{15}$$

where $\varrho(\mathbf{z}, \widehat{\mathbf{z}}, v, w)$ accounts for the higher order terms. Recall that $\mathbf{r}_i = [d_i \ \vartheta_i]^{\mathrm{T}}$ with $d_i \in (0, \infty)$ and $\vartheta_i \in [-\pi, \pi)$ for all $t$. Then it is clear that the following bound holds

$$\|\mathbf{A}(t)\| = \overline{a} < \infty \tag{16}$$

for all $t$ where for any time-varying matrix $\mathbf{M}(t)$ we assume the following

$$\|\mathbf{M}(t)\| = \sup\{\|\mathbf{M}(t)\| : m_{ij} \in \mathbb{M}_{ij} \subseteq \mathbb{R}\} \tag{17}$$

for all $t$ and for some norm $\|\cdot\|$. At this point we make the following assumptions.

*Assumption 1: The translational velocity of the robot $v(t)$ is uppperbounded in any arbitrary coordinate scale such that $v(t) \leq \overline{v}$ for all $t$. For simplicity we also assume that $v(t) > 0$ for all $t$. Now it follows that there exists a temporal coordinate scale such that $v(t) \leq 1$ for all $t$.*

*Assumption 2: The relative distance between the robot and the $i^{th}$ landmark at time $t$ belongs to the space $d_i(t) \in (0, \infty)$ in any arbitrarily chosen coordinate scale. There exists a spatial coordinate scale such that for all $t$ we have $d_i(t) \in [1, \infty)$.*

Assumptions 1 and 2 are weak (actually notational) and can almost surely be satisfied in practice (i.e. by finding explicit spatial and temporal scales). The case of $v = 0$ is trivially obtained from the subsequent results. For simplicity we also assume the following.

*Assumption 3: For all $t$ we have $\widehat{\mathbf{r}}_i(t) = [\widehat{d}_i(t) \ \widehat{\vartheta}_i(t)]^{\mathrm{T}}$ with $\widehat{d}_i(t) \in [1, \infty)$ and $\widehat{\vartheta}_i(t) \in [-\pi, \pi)$.*

Assumption 3 calls for the state estimate components to be restricted to the assumed true global state space. Finally, we make the following assumption on the design parameters.

*Assumption 4: The following $\mathbf{Q}(t) \geq \underline{q}\mathbf{I}$, $\mathbf{R}(t) \geq \underline{r}\mathbf{I}$ and $\mathbf{P}(t_0) \geq p_0\mathbf{I}$ are given for some $\underline{q}, \underline{r}, p_0 > 0$ such that $\|\mathbf{Q}(t)\| \geq \underline{q}$ and $\|\mathbf{R}(t)\| \geq \underline{r}$. Moreover, $\mathbf{Q}(t)$ and $\mathbf{R}(t)$ are chosen to be bounded by $\|\mathbf{Q}(t)\| \leq \overline{q} < \infty$ and $\|\mathbf{R}(t)\| \leq \overline{r} < \infty$ for all $t$. Also, we have $\|\mathbf{E}(t)\| \leq \overline{e} < \infty$ with $\mathbf{E}(t) \geq \underline{e}\mathbf{I}$.*

Clearly, Assumption 4 is standard. We will also need the following lemma concerning the growth of $\varrho(\mathbf{z}, \widehat{\mathbf{z}}, v_r, w_r)$.

*Lemma 1: The following inequality holds*

$$\begin{aligned} \|\varrho(\mathbf{z}, \widehat{\mathbf{z}}, v_r, w_r)\| \quad &= \quad \|f(\mathbf{z}, \cdot) - f(\widehat{\mathbf{z}}, \cdot) - \mathbf{A}(t)\left(\mathbf{z} - \widehat{\mathbf{z}}\right)\| \\ &\leq \quad 2\overline{a}\|\boldsymbol{\zeta}\| \end{aligned} \tag{18}$$

*for $|\mathcal{V}| = n$ with probability 1 when Assumptions 1-4 hold.*

*Proof:* The proof is trivial and follows from (16) and the triangle inequality. ■

Note also that $\varrho(\mathbf{z}, \widehat{\mathbf{z}}, v_r, w_r) = 0$ when $\boldsymbol{\zeta}(t) = 0$. We assume the initial estimation error $\boldsymbol{\zeta}(t_0)$ belongs to the set

$$\boldsymbol{\zeta}(t_0) \in \{\boldsymbol{\eta} \in \{[0, \infty) \times [-\pi, \pi)\} : \|\boldsymbol{\zeta}(t_0)\| \leq d\} \tag{19}$$

for some constant $d < \infty$. We also assume initially that $\mathcal{G}(t) = \mathcal{V}$ for all $t > t_0$ and thus the error propagates according to (14) with (for simplicity) $\mathbf{H}(\mathcal{G}(t)) = \mathbf{I}$ for all $t$. It is common to assume a full measurement vector when performing such an analysis [8].

*Lemma 2:* Suppose Assumptions 1-4 hold. Then the state estimate covariance $\mathbf{P}(t)$ is bounded by

$$0 < \underline{p} \leq \|\mathbf{P}(t)\| \leq \overline{p} < \infty \tag{20}$$

for all $t > t_0$ and where

$$\overline{p} \triangleq \left(\|\mathbf{P}(t_0)\| + \frac{\|\mathbf{Q}(t)\| + \|\mathbf{R}(t)\|\|\boldsymbol{\Lambda}(t)\|^2}{2\kappa}\right) \tag{21}$$

and where $\mathbf{\Lambda}$ is chosen such that

$$\boldsymbol{\eta}^{\mathrm{T}}\left(\mathbf{A}(t)+\mathbf{\Lambda}(t)\right)\boldsymbol{\eta} \leq -\kappa\|\boldsymbol{\eta}\|^2 \tag{22}$$

is satisfied for all $\boldsymbol{\eta} \in \mathbf{R}^2$ with $\kappa > 0$.

*Proof:* The upper bound can be obtained by considering the following time-varying linear control system

$$-\dot{\mathbf{q}} = \mathbf{A}(t)\mathbf{q} + \mathbf{u} \tag{23}$$

with a boundary $\mathbf{q}(T) = \mathbf{q}_T$ for some $\infty \geq T > 0$ and with controllability Grammian

$$\mathcal{C}(t+\tau,t) = \int_t^{t+\tau} \mathbf{\Phi}(t+\tau,t)\mathbf{\Phi}^{\mathrm{T}}(t+\tau,t)dt \tag{24}$$

where $\mathbf{\Phi}(t+\tau,t)$ is the fundamental matrix with $\mathbf{\Phi}(t,t) = \mathbf{I}$. Clearly, the system (23) is uniformly completely controllable. Consider the following cost function

$$\mathcal{J}(t,\tau,\mathbf{q},\mathbf{u}) = \mathcal{B}(t_0,\mathbf{q}(t_0)) + \int_{t_0}^{T}\left(\mathbf{q}^{\mathrm{T}}\mathbf{Q}\mathbf{q} + \mathbf{u}^{\mathrm{T}}\mathbf{R}\mathbf{u}\right)dt \tag{25}$$

and value function $\mathcal{B}(t,\mathbf{q}(t)) = \mathbf{q}^{\mathrm{T}}(t)\mathbf{P}(t)\mathbf{q}(t)$. Let the control input equal $\mathbf{u}(t) = \mathbf{\Lambda}(t)\mathbf{q}$ for some continuous bounded matrix $\mathbf{\Lambda}(t)$ such that $-\dot{\mathbf{q}} = (\mathbf{A}(t)+\mathbf{\Lambda}(t))\mathbf{q}$. Note now that

$$\begin{aligned}\mathcal{B}(T,\mathbf{q}(T)) &= \mathbf{q}^{\mathrm{T}}(T)\mathbf{P}(T)\mathbf{q}(T)\\ &\leq \mathcal{B}(t_0,\mathbf{q}(t_0)) + \\ &\quad \int_{t_0}^{T}\mathbf{q}^{\mathrm{T}}\left(\mathbf{Q}+\mathbf{\Lambda}^{\mathrm{T}}(t)\mathbf{R}\mathbf{\Lambda}(t)\right)\mathbf{q}dt\end{aligned} \tag{26}$$

Solving $-\dot{\mathbf{q}} = (\mathbf{A}(t)+\mathbf{\Lambda}(t))\mathbf{q}$ for $\mathbf{q}(T)$ implies that

$$\begin{aligned}\|\mathbf{q}(T)\|^2 &= \|\mathbf{q}_T\|^2 = \|\mathbf{q}(t_0)\|^2 - \\ &\quad 2\int_{t_0}^{T}\mathbf{q}^{\mathrm{T}}\left(\mathbf{A}(t)+\mathbf{\Lambda}(t)\right)\mathbf{q}\;dt\end{aligned} \tag{27}$$

and (22) implies $\|\mathbf{q}(t_0)\|^2 \leq \|\mathbf{q}_T\|^2$ and $\int_{t_0}^{T}\mathbf{q}^{\mathrm{T}}\mathbf{q}\;dt \leq \frac{\|\mathbf{q}_T\|^2}{2\kappa}$. Using this with (26) leads to the upper-bound. ■

Note that $\|\mathbf{P}(t)\|$ is bounded above by a constant independent of the time $t > t_0$. Part of Lemma 2 follows from a theorem given in [9]. The condition (22) calls for the system pair $\mathbf{A}(t)$ and $\mathbf{H}(\mathcal{G}(t))$ to be uniformly detectable. In our case we know that the system is observable (which implies detectability [9], [10]). As such, a suitable matrix $\mathbf{\Lambda}(t)$ exists with probability one.

*Theorem 1:* Consider the system (14) with an initial condition (19) and $\mathbf{H}(\mathcal{G}(t)) = \mathbf{I}$. Suppose that Assumptions 1-4 hold. If $\|\mathbf{P}^{-1}(t)\mathbf{Q}(t)\mathbf{P}^{-1}(t) + \mathbf{R}^{-1}(t)\|\underline{p} > \frac{4\overline{a}\overline{p}}{\underline{p}}$ then the estimation error is bounded above with

$$\mathcal{E}\{\|\boldsymbol{\zeta}(t)\|^2\} \leq \max\left\{\frac{n\overline{p}^2\bar{e}^2}{2\gamma\underline{r}^2}\;,\;\frac{\overline{p}}{\underline{p}}\|\boldsymbol{\zeta}(t_0)\|^2\right\} \tag{28}$$

where $\gamma = \|\mathbf{P}^{-1}(t)\mathbf{Q}(t)\mathbf{P}^{-1}(t) + \mathbf{R}^{-1}(t)\|\underline{p} - 4\overline{a}\overline{p}/\underline{p}$ and the error $\mathcal{E}\{\|\boldsymbol{\zeta}(t)\|^2\}$ as $t \to \infty$ is bounded by $\frac{n\overline{p}^2\bar{e}^2}{2\gamma\underline{r}^2}$.

*Proof:* The error system (14) can be thought of as a linear system with a nonlinear perturbation being driven by a zero-mean Weiner process. Let $\mathcal{B}(t,\boldsymbol{\zeta}(t)) = \boldsymbol{\zeta}^{\mathrm{T}}(t)\mathbf{P}^{-1}(t)\boldsymbol{\zeta}(t) > 0$ and note that

$$\begin{aligned}d\mathcal{B} &= \left[\frac{\partial\mathcal{B}}{\partial t} + \frac{\partial\mathcal{B}}{\partial\boldsymbol{\zeta}}\left(\mathbf{A}(t)-\mathbf{K}(t)\right)\boldsymbol{\zeta}\right]dt + \\ &\quad \frac{\partial\mathcal{B}}{\partial\boldsymbol{\zeta}}\varrho(\mathbf{z},\widehat{\mathbf{z}},v_r,w_r)dt + \\ &\quad \frac{1}{2}\mathrm{tr}\left(\mathrm{hess}(\mathcal{B})\mathbf{K}(t)\mathbf{E}(t)\mathbf{E}^{\mathrm{T}}(t)\mathbf{K}^{\mathrm{T}}(t)\right)dt - \\ &\quad \frac{\partial\mathcal{B}}{\partial\boldsymbol{\zeta}}\mathbf{K}(t)\mathbf{E}(t)d\mathbf{n}\\ d\mathcal{B} &= \left[\frac{\partial\mathcal{B}}{\partial t} + \mathcal{L}\mathcal{B}\right]dt - \frac{\partial\mathcal{B}}{\partial\boldsymbol{\zeta}}\mathbf{K}(t)\mathbf{E}(t)d\mathbf{n}\end{aligned} \tag{29}$$

using Ito's differential formula and where $\mathcal{L}$ is the Kolmogorov backward operator, $\mathrm{hess}(\cdot)$ denotes the Hessian operator and $\mathrm{tr}(\cdot)$ denotes the matrix trace. Evaluating the terms and re-arranging leads to

$$\begin{aligned}d\mathcal{B} &= \left[\boldsymbol{\zeta}^{\mathrm{T}}\left[-\mathbf{P}^{-1}(t)\mathbf{Q}(t)\mathbf{P}^{-1}(t) - \mathbf{R}^{-1}(t)\right]\boldsymbol{\zeta}\right]dt + \\ &\quad 2\boldsymbol{\zeta}^{\mathrm{T}}\mathbf{P}^{-1}(t)\varrho(\mathbf{z},\widehat{\mathbf{z}},v_r,w_r)dt + \\ &\quad \frac{1}{2}\mathrm{tr}\left(\mathbf{R}^{-1}(t)\mathbf{E}(t)\mathbf{E}^{\mathrm{T}}(t)\mathbf{R}^{-\mathrm{T}}(t)\mathbf{P}^{\mathrm{T}}(t)\right)dt - \\ &\quad 2\boldsymbol{\zeta}^{\mathrm{T}}\mathbf{R}^{-1}(t)d\mathbf{n}\\ &\leq \left[-\alpha\|\boldsymbol{\zeta}\|^2 + \frac{4\overline{a}}{\underline{p}}\|\boldsymbol{\zeta}\|^2 + \frac{n\overline{p}\bar{e}^2}{2\underline{r}^2}\right]dt - \\ &\quad 2\boldsymbol{\zeta}^{\mathrm{T}}\mathbf{R}^{-1}(t)d\mathbf{n}\end{aligned} \tag{30}$$

where we have explicitly employed Lemma 1 and Lemma 2 and where

$$\alpha = \|\mathbf{P}^{-1}(t)\mathbf{Q}(t)\mathbf{P}^{-1}(t) + \mathbf{R}^{-1}(t)\| \tag{31}$$

Clearly we have $\overline{p}^{-1}\|\boldsymbol{\zeta}\|^2 \leq \mathcal{B}(t,\boldsymbol{\zeta}(t)) \leq \underline{p}^{-1}\|\boldsymbol{\zeta}\|^2$ such that some simple algebra implies that

$$\begin{aligned}d\mathcal{B} &\leq -\left(\alpha\underline{p} - \frac{4\overline{a}\overline{p}}{\underline{p}}\right)\mathcal{B}dt + \frac{n\overline{p}\bar{e}^2}{2\underline{r}^2}dt - \\ &\quad 2\boldsymbol{\zeta}^{\mathrm{T}}\mathbf{R}^{-1}(t)d\mathbf{n}\\ \mathcal{B} &\leq \mathcal{B}(t_0,\boldsymbol{\zeta}(t_0)) - \\ &\quad \int_{t_0}^{t}\left(\alpha\underline{p} - \frac{4\overline{a}\overline{p}}{\underline{p}}\right)\mathcal{B}(\tau,\boldsymbol{\zeta}(\tau))d\tau + \\ &\quad \frac{n\overline{p}\bar{e}^2}{2\underline{r}^2}\int_{t_0}^{t}d\tau - 2\int_{t_0}^{t}\boldsymbol{\zeta}^{\mathrm{T}}(\tau)\mathbf{R}^{-1}(\tau)d\mathbf{n}(\tau)\end{aligned} \tag{32}$$

From the Bellman-Gromwall lemma [11] we have

$$\begin{aligned}\mathcal{B}(t,\boldsymbol{\zeta}(t)) &\leq \mathcal{B}(t_0,\boldsymbol{\zeta}(t_0))\exp\left(-\gamma(t-t_0)\right) + \\ &\quad \frac{n\overline{p}\bar{e}^2}{2\gamma\underline{r}^2}\left(1-\exp\left(-\gamma(t-t_0)\right)\right) - \\ &\quad 2\int_{t_0}^{t}\boldsymbol{\zeta}^{\mathrm{T}}(\tau)\mathbf{R}^{-1}(\tau)d\mathbf{n}(\tau)\end{aligned} \tag{33}$$

where

$$\gamma = \left(\alpha\underline{p} - 4\overline{a}\overline{p}/\underline{p}\right) \tag{34}$$

with $\gamma > 0$ if and only if $\alpha \underline{p} > \frac{4\overline{a}\overline{p}}{\underline{p}}$. Taking the expectation $\mathcal{E}\{\cdot\}$ of both sides of (33) gives

$$\begin{aligned}\mathcal{E}\{\mathcal{B}(t,\boldsymbol{\zeta}(t))\} &\leq \mathcal{B}(t_0,\boldsymbol{\zeta}(t_0))\exp\left(-\gamma(t-t_0)\right) + \\ &\quad \frac{n\overline{p}\overline{e}^2}{2\gamma\underline{r}^2}\left(1-\exp\left(-\gamma(t-t_0)\right)\right) \quad (35)\end{aligned}$$

and thus

$$\begin{aligned}\mathcal{E}\{\|\boldsymbol{\zeta}(t)\|^2\} &\leq \frac{\overline{p}}{\underline{p}}\|\boldsymbol{\zeta}(t_0)\|^2\exp\left(-\gamma(t-t_0)\right) + \\ &\quad \frac{n\overline{p}^2\overline{e}^2}{2\gamma\underline{r}^2}\left(1-\exp\left(-\gamma(t-t_0)\right)\right) \quad (36)\end{aligned}$$

We then easily find that

$$\mathcal{E}\{\|\boldsymbol{\zeta}(t)\|^2\} \leq \max\left\{\frac{n\overline{p}^2\overline{e}^2}{2\gamma\underline{r}^2}\ ,\ \frac{\overline{p}}{\underline{p}}\|\boldsymbol{\zeta}(t_0)\|^2\right\} \quad (37)$$

for all $t$ if $\gamma > 0$ and the error $\mathcal{E}\{\|\boldsymbol{\zeta}(t)\|^2\}$ as $t \to \infty$ is bounded by $\frac{n\overline{p}^2\overline{e}^2}{2\gamma\underline{r}^2}$. This completes the proof. ■

Importantly, we have shown under what conditions an EKF-like algorithm will yield an exponentially bounded and converging mean-square estimation error. The asymptotic mean-square estimation error is dependent on the specific robot trajectory but is upper-bounded by $\frac{n\overline{p}^2\overline{e}^2}{\gamma\underline{r}^2}$. Theorem 1 is a significant contribution to the problem of localization using a mobile sensor and is a fundamental result.

*Corollary 2: Suppose that Assumptions 1-4 hold and* $(\mathbf{A}, \mathbf{H}(t))$ *is a uniformly detectable pair (which is guaranteed since* $(\mathbf{A}, \mathbf{H}(t))$ *is actually an observable pair). Now if* $\gamma > 0$ *and* $\mathbf{n} \to \mathbf{0}$*, then* $\|\boldsymbol{\zeta}(t)\| \to 0$ *as* $t \to \infty$*.*

That is, as the measurement noise approaches zero, the estimation error will asymptotically (and actually exponentially [9], [12]) converge to zero given the satisfaction of the required conditions; i.e. the EKF as applied in this paper acts as an asymptotic nonlinear observer; e.g. see [7], [9], [12]–[14]. Thus, Corollary 2 and Theorem 1 justify application of the EKF in well-posed scenarios (where the noise is small). We can also derive a result similar to Theorem 1 when process noise (i.e. control input noise) is present. For brevity and due to space limitations, estimator simulations will appear in an extended version of the paper.

## IV. Active Sensor-Object Pose Control

We now illustrate a technique to steer the sensor to a desired relative sensor-object pose $\mathbf{t}_i = [d_{ti}\ \vartheta_{ti}]^{\mathrm{T}}$ using a simple continuous control law; e.g. similarly to the formation control problem [15]. This might be desired if the mobile sensor wishes to view (with a visual sensor for example) a particular object $i \in \mathcal{V}$ from a (possibly estimated) distance and viewing angle. Similarly, the mobile sensor might be a robot which must achieve a certain robot-object pose in order to manipulate the object in some manner (due to the physical configuration or constraints of the manipulation device).

Consider the global Cartesian sensor motion equations (1) and a Cartesian representation of the $i^{th}$ object's position. Steering the sensor to achieve a desired distance $d_i$ and a desired relative (viewing) angle $\vartheta_i$ with the object is non-trivial since the desired objective is not stated linearly in the sensor state components. Moreover, it would require a discontinuous or time-varying nonlinear control law.

However, consider the relative state $\mathbf{r}_i = [d_i\ \vartheta_i]^{\mathrm{T}}$ and the problem of steering the mobile sensor to a desired relative state $\mathbf{t}_i = [d_{ti}\ \vartheta_{ti}]^{\mathrm{T}}$. Note that the control objective is expressed naturally and the sensor state-object state is linearly related to the objective.

The described polar formulation also has a very attractive property in that we can use Lyupunov techniques to design the stabilizing sensor-object pose control law. Brockett's (negative) theorem is in a sense circumvented (albeit we do not control the robot pose in a global sense) and the practicality is (arguably) increased by considering such a formulation. The controller we outline is continuous and leads to very natural trajectories.

We now outline the control law for $v$ and $w$ that will steer the mobile sensor to have a desired (or target) pose $\mathbf{t}_i = [d_{ti}\ \vartheta_{ti}]^{\mathrm{T}}$ with respect to the estimated state of object $i$ given by $\widehat{\mathbf{r}}_i(\tau) = [\widehat{d}_i(\tau)\ \widehat{\vartheta}_i(\tau)]^{\mathrm{T}}$ at some time $\tau$. The following remark concerns an implicit technical requirement of the controller with respect to the considered estimation problem outlined in the previous section.

*Remark 1: We have a state subspace estimate* $\widehat{\mathbf{r}}_i(\tau) = [\widehat{d}_i(\tau)\ \widehat{\vartheta}_i(\tau)]^{\mathrm{T}}$ *at some time* $\tau$ *as the output from the EKF algorithm discussed in the previous section. Now we can set to zero the Kalman gain* $\mathbf{K}(t)$ *subspace corresponding to the state* $\widehat{\mathbf{r}}_i$ *of object* $i$ *for all* $t > \tau$*. Then we have measurements (or estimates as it so happens) of the relative sensor object pose* $\widehat{\mathbf{r}}_i(t) = [\widehat{d}_i(t)\ \widehat{\vartheta}_i(t)]^{\mathrm{T}}$ *for all* $t > \tau$ *that are not affected by a stochastic process* $\forall t > \tau$*. For example, if the sensor does not move such that* $v = 0$ *and* $w = 0$ *then* $\widehat{\mathbf{r}}_i(t) = [\widehat{d}_i(t)\ \widehat{\vartheta}_i(t)]^{\mathrm{T}}$ *for all* $t > \tau$ *is constant. This does not necessarily occur when the Kalman gain* $\mathbf{K}(t)$ *subspace corresponding to the state* $\widehat{\mathbf{r}}_i$ *is non-zero and we are taking measurements of object* $i$*.*

Thus we want the control error

$$\boldsymbol{\delta}_{ti}(t) = \mathbf{t}_i - \widehat{\mathbf{r}}_i(t) = [d_{ti}\ \vartheta_{ti}]^{\mathrm{T}} - [\widehat{d}_i(t)\ \widehat{\vartheta}_i(t)]^{\mathrm{T}},\ \ t > \tau \quad (38)$$

to be minimized to zero where $\widehat{\mathbf{r}}_i(t)$ is the subspace output of the EKF-like algorithm given that we have set to zero the Kalman gain $\mathbf{K}(t)$ subspace corresponding to the state $\widehat{\mathbf{r}}_i$ of object $i$ for all $t > \tau$. The following theorem outlines the control law and states the stability result.

*Theorem 2:* Consider the control error (38) and suppose that Assumptions 1-4 hold. The control inputs are given by

$$\begin{aligned} v &= -k_1\cos\left(\widehat{\vartheta}_i(t)\right)\left(d_{ti} - \widehat{d}_i(t)\right) \\ w &= -k_2\left(\vartheta_{ti} - \widehat{\vartheta}_i(t)\right),\ \ \forall t > \tau \end{aligned} \quad (39)$$

where Assumption 3 specifies $\widehat{\vartheta}_i(t) \in [-\pi, \pi)$ and $k_2 \geq k_1 \geq 1$ are control gains. Assume that $\vartheta_{ti} \neq \pm\frac{\pi}{2}$. Then the error (38) asymptotically and exponentially converges to zero given any initial sensor-object configuration $\widehat{\mathbf{r}}_i(\tau) = [\widehat{d}_i(\tau)\ \widehat{\vartheta}_i(\tau)]^{\top}$.

*Proof:* The error $\boldsymbol{\delta}_{ti}(t)$ obeys

$$\dot{\delta}_{ti}(t) = \begin{bmatrix} -k_1 \cos^2 \widehat{\vartheta}_i(t) & 0 \\ -\frac{k_1}{\widehat{d}_i(t)} \sin \widehat{\vartheta}_i(t) \cos \widehat{\vartheta}_i(t) & -k_2 \end{bmatrix} \delta_{ti}(t) \quad (40)$$

for $t > \tau$ and where Assumption 2 claims there exists a coordinate scale such that $d_i(t) \geq 1$ in any practical scenario. We also have Assumption 3 which claims the estimated state output will belong to the adopted state space such that $\widehat{d}_i(t) \geq 1$. Note that differential equation (40) is of the form $\dot{\delta}_{ti}(t) = \mathbf{F}(\delta_{ti})\delta_{ti}(t)$ and is nonlinear since $\mathbf{F}(\delta_{ti})$ is dependent on the error. Let $\mathcal{B}(\delta_{ti}(t)) = \delta_{ti}(t)^\top \delta_{ti}(t)$ be a candidate Lyapunov function. It remains to establish that $\mathbf{F}(\delta_{ti}) + \mathbf{F}(\delta_{ti})^\top$ is negative definite. If $\widehat{\vartheta}_i(t) \neq \pm\frac{\pi}{2}$ then under the adopted assumptions it is easy to verify

$$\mathrm{tr}(\mathbf{F}(\delta_{ti}) + \mathbf{F}(\delta_{ti})^\top) < 0 \quad (2)$$

$$\det(\mathbf{F}(\delta_{ti}) + \mathbf{F}(\delta_{ti})^\top) > 0 \quad (3)$$

If $\vartheta_{ti} \neq \pm\frac{\pi}{2}$ is not a desired pose objective then clearly $\dot{\delta}_{ti}(t)$ is not at equilibrium and $w \neq 0$. Thus $\widehat{\vartheta}_i(t) \neq \pm\frac{\pi}{2}$ represent non-attractive and non-invariant manifolds in the state space. This completes the proof. ■

A relative angle $\vartheta_{ti} = \pm\pi/2 \pm \epsilon$ for any arbitrarily small $\epsilon > 0$ is stabilizable given the designed continuous controller. In practice this is quite sufficient. To achieve an exact relative angle $\vartheta_{ti} = \pm\pi/2$ requires a slight (technical) modification of the control law for $v$ and is straightforward but results in a control function for $v$ that is discontinuous at $\widehat{\vartheta}_i(t) = \pm\frac{\pi}{2}$. The details are omitted for brevity but are quite simple.

We thus have illustrated how a polar formulation of the problems considered can be directly exploited to yield very simple solutions in a very natural form. The controlled sensor trajectories are also very natural. We now consider an example involving a robot with unicycle kinematics (1) and initial state $\mathbf{s} = [0\ 0\ 0]^{\mathrm{T}}$. We have randomly placed an object (simulating a random initial sensor-object pose) in the environment. The desired relative pose is characterized solely by $\mathbf{t}_i = [2\ \ -\pi/4]^{\mathrm{T}}$ and $k_1 = k_2 = 0.2$. Figure 1 part (a) illustrates the sensor trajectory and part (b) illustrates the range and angle error convergence.

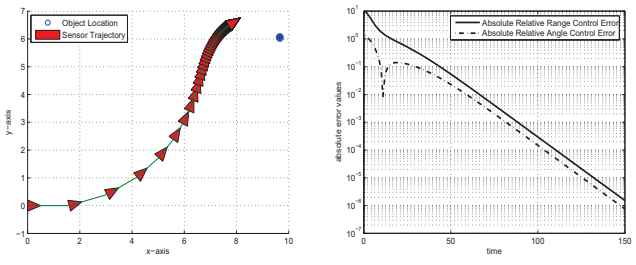



Fig. 1. *(a) shows the sensor trajectory and (b) shows the error convergence.*

From Figure 1 we note the natural and continuous sensor trajectory and the fast convergence of the errors to zero. Nothing more than Lyapunov methods were used to prove control stability. Additional examples are omitted for brevity.

## V. Concluding Remarks

The problem of object localization using a mobile sensor was examined in this paper. We derived a coordinate transform and a relative sensor-object motion model that leads to a novel problem formulation where the measurements are linear in the object positions. We then apply an extended Kalman filter-like algorithm to the estimation problem. Using stochastic calculus we analyzed the convergence properties of the filter. We then illustrate that it is possible to steer the mobile sensor back to a relative sensor-object pose using a simple continuous control law. This last fact is significant since we can circumvent Brockett's negative result. The polar formulation considered in this paper provides a very natural representation of the general localization and sensor-object pose control problems. This simplifies the design of the filter and the control law (since the actual problem is represented naturally and so are the control objectives) and it also improves the performance of the estimator (as no approximate linearization of the measurements is needed).

## References


[1] V. Aidala and S. Hammel. Utilization of modified polar coordinates for bearings-only tracking. *IEEE Transactions on Automatic Control*, 28(3):283–294, March 1983.

[2] M. Cao and A.S. Morse. The use of dwell-time switching to maintain a formation with only range sensing. In *Proceedings of the 3rd International Symposium on Communications, Control and Signal Processing (ISCCSP)*, pages 954–959, March 2008.

[3] A.N. Bishop, B. Fidan, B.D.O. Anderson, K. Dogancay, and P.N. Pathirana. Optimality analysis of sensor-target geometries in passive localization: Part 1 - Bearing-only localization. In *Proceedings of the 3rd International Conference on Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP)*, December 2007.

[4] A.N. Bishop, B. Fidan, B.D.O. Anderson, P.N. Pathirana, and K. Dogancay. Optimality analysis of sensor-target geometries in passive localization: Part 2 - Time-of-arrival based localization. In *Proceedings of the 3rd International Conference on Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP)*, December 2007.

[5] J. Aloimonos, I. Weiss, and A. Bandyopadhyay. Active vision. *International Journal of Computer Vision*, 1(4):333–356, January 1988.

[6] R.M. Murray, Z. Li, and S.S. Sastry. *A Mathematical Introduction to Robotic Manipulation*. CRC Press, New York, NY, 1994.

[7] T. Song and J. Speyer. A stochastic analysis of a modified gain extended Kalman filter with applications to estimation with bearings only measurements. *IEEE Transactions on Automatic Control*, 30(10):940–949, October 1985.

[8] M.W.M.G. Dissanayake, P. Newman, S. Clark, H.F. Durrant-Whyte, and M. Csorba. A solution to the simultaneous localization and map building (SLAM) problem. *IEEE Transactions on Robotics and Automation*, 17(3):229–241, June 2001.

[9] J.S. Baras, A. Bensoussan, and M.R. James. Dynamic observers as asymptotic limits of recursive filters: Special cases. *SIAM Journal on Applied Mathematics*, 48(5):1147–1158, October 1988.

[10] B.D.O. Anderson and J.B. Moore. *Optimal Control: Linear Quadratic Methods*. Prentice-Hall, Englewood Cliffs, N.J., 1989.

[11] S. Sastry. *Nonlinear Systems: Analysis, Stability and Control*. Springer-Verlag, New York, N.Y., 1999.

[12] K. Reif and R. Unbehauen. The extended Kalman filter as an exponential observer for nonlinear systems. *IEEE Transactions on Signal Processing*, 47(8):2324–2328, August 1999.

[13] K. Reif, S. Gunther, E. Yaz, and R. Unbehauen. Stochastic stability of the discrete-time extended Kalman filter. *IEEE Transactions on Automatic Control*, 44(4):714–728, April 1999.

[14] K. Reif, S. Gunther, E. Yaz, and R. Unbehauen. Stochastic stability of the continuous-time extended Kalman filter. *IEE Proceedings - Control Theory and Applications*, 147(1):45–52, January 2000.

[15] B.D.O Anderson, C. Yu, S. Dasgupta, and A.S. Morse. Control of a three-coleader formation in the plane. *Systems and Control Letters*, 56(9-10):573–578, September-October 2007.

2009 IEEE International Conference on Robotics and Automation
Kobe International Conference Center
Kobe, Japan, May 12-17, 2009

# A Stochastically Stable Solution to the Problem of Robocentric Mapping

Adrian N. Bishop and Patric Jensfelt

*Abstract*— **This paper provides a novel solution for robocentric mapping using an autonomous mobile robot. The robot dynamic model is the standard unicycle model and the robot is assumed to measure both the range and relative bearing to the landmarks. The algorithm introduced in this paper relies on a coordinate transformation and an extended Kalman filter like algorithm. The coordinate transformation considered in this paper has not been previously considered for robocentric mapping applications. Moreover, we provide a rigorous stochastic stability analysis of the filter employed and we examine the conditions under which the mean-square estimation error converges to a steady-state value.**

## I. Introduction

Simultaneous localization and mapping (or SLAM) refers to the process of building a map of an environment from sensory information gathered by a mobile robot, while simultaneously estimating the position of the robot using the map [1]–[6]. An introduction to the SLAM problem is available in many papers; e.g. see [7], [8] and the references therein for an overview of the different approaches. Following the work in [1], one of the most common methods for solving the SLAM problem is to use an extended Kalman Filter (EKF). However, the traditional SLAM state vector[1] [1], [2], [4] in a global coordinate system is not observable as discussed in [9] given only relative landmark-robot measurements such range and/or bearing. Another problem is that of estimator inconsistencies caused by accumulated linearization errors [10]–[12]. In [13] the concept of robocentric mapping is introduced and this concept it is shown to better deal with linearization errors than the traditional SLAM formulation.

The EKF consistency and the convergence of the approximate EKF covariance matrix is analyzed in [12] for the general problem of SLAM. However, it is possible, in the framework of the EKF, for the covariance matrix to be asymptotically bounded while the state estimation error diverges asymptotically. Moreover, it is the state estimate itself that will be used by the robot when making decisions etc. Hence, the actual (or mean) estimation error is a more meaningful quantity to analyze.

The primary contribution of this paper is the development of a robocentric mapping algorithm based on a simple, yet particularly important, coordinate transformation. By building a map in a relative polar framework we eliminate the nonlinearities associated with the measurement equation. Moreover, we eliminate the difficulties associated with the unobservable states [9] and the inconsistencies caused by the affect of the EKF linearizations (which alter the unobservable subspace [9]). A robocentric map is also (arguably) more useful/natural than a global map for a large class of problems. The relative robot dynamic model remains nonlinear but takes on a different form. We then apply the standard extended Kalman filter (EKF) to this problem and justify this approach via a rigorous stochastic convergence analysis. The convergence of the EKF relative map is given in terms of the mean estimation error and is based on stochastic calculus. The convergence analysis in this paper is necessarily conservative, with the particular asymptotic properties of the mean estimation error being naturally dependent on the exact robot trajectory; e.g. see [14]–[18].

The approach and analysis given in this paper was partly inspired by [9][13] where the difficulties of the global SLAM problem are highlighted and where it is implied (perhaps not always explicitly) that a robocentric approach would circumvent many of these problems. The coordinate framework chosen in this paper was inspired by the large bearing-only tracking literature where it is shown that removing the nonlinearities associated with the measurement equation can significantly improve the EKF performance [19], [20]. Finally, a rigorous mean-error convergence analysis was given to further justify the application of the EKF and to provide a deeper insight into the proposed robocentric mapping algorithm.

The remainder of this paper is organized as follows. In Section II we introduce some preliminary notation and conventions. In Section III we introduce the concept of mapping in polar coordinates using a robocentric framework. We outline the standard extended Kalman filter-like algorithm which forms the basis of the estimator considered in this paper. In Section III we then analyze the observability of the mapping problem considered and the convergence properties of the particular estimator considered. In Section IV we present some simple simulation results and in Section V we relate our robocentric problem to the traditional SLAM problem. In Section VI we give our conclusions.

## II. Preliminaries

Consider a single robot with a state $\mathbf{s}_r = [x_r \ y_r \ \phi_r]^\top \in \{\mathbb{R}^2 \times \mathbb{SO}(1,\mathbb{R})\}$ where $x_r$ and $y_r$ are the robot's Cartesian position coordinates and $\phi_r$ is the robot's heading. The robot dynamics are based on the unicycle model,

$$\begin{aligned} \dot{x}_r &= v_r \cos\phi_r \\ \dot{y}_r &= v_r \sin\phi_r \\ \dot{\phi}_r &= w_r \end{aligned} \tag{1}$$

A.N. Bishop and P. Jensfelt are with the Centre for Autonomous Systems, KTH, Stockholm, Sweden. This work was supported by the Centre for Autonomous Systems (CAS) and the EU FP7 project CogX.

[1]The traditional SLAM state vector consists of the pose of the robot and the Cartesian location of the landmarks.

where $v_r$ is the translational velocity and $w_r$ is the robots angular velocity. Note that there are three robot state variables in $\{\mathbb{R}^2 \times \mathbb{SO}(1,\mathbb{R})\}$ and only two control inputs. The nonholonomic constraint on the robot is given by

$$\dot{x}_r \sin\phi_r = \dot{y}_r \cos\phi_r \tag{2}$$

The robot will generally only know $v_r$ and $w_r$ up to some error denoted by $v$ and $w$ respectively. Here, $v$ and $w$ are assumed to be uncorellated zero-mean Weiner processes. The dynamics of the robot are thus assumed to obey

$$d\begin{bmatrix} x_r \\ y_r \\ \phi_r \end{bmatrix} = \begin{bmatrix} v_r\cos\phi_r \\ v_r\sin\phi_r \\ w_r \end{bmatrix} dt + \begin{bmatrix} \sigma_v\cos\phi_r & 0 \\ \sigma_v\sin\phi_r & 0 \\ 0 & \sigma_w \end{bmatrix} \begin{bmatrix} dv \\ dw \end{bmatrix} \tag{3}$$

which is a stochastic differential equation of the Ito-type. Here, $\sigma_v$ and $\sigma_w$ are the standard-deviations of the errors $v$ and $w$ respectively. The environment is populated with a set $\mathcal{V}$ of landmark (or feature) points with $|\mathcal{V}| = n$. The Cartesian position of the $i^{th}$ landmark is denoted by $\mathbf{p}_i = [x_i \ y_i]^\top \in \mathbb{R}^2$. The landmarks are stationary in this case and represent the map of the environment which is to be estimated by the mobile robot. At some time $t$ the robot can sense a subset $\mathcal{G}(t) \subseteq \mathcal{V}$ of landmarks. At time $t$ the true robot measurements are given by

$$\begin{aligned} d_i &= \sqrt{(x_i - x_r)^2 + (y_i - y_r)^2} \\ \vartheta_i &= \theta_i - \phi_r = \arctan\left(\frac{y_i - y_r}{x_i - x_r}\right) - \phi_r \\ &\quad \forall i \in \mathcal{G}(t) \end{aligned} \tag{4}$$

where $\vartheta_i = \theta_i - \phi_r$ is the relative bearing to $i^{th}$ landmark in the robots internal Cartesian coordinate system, i.e. the Cartesian coordinate system rotated by the robots heading. Let $\mathbf{z} = [\mathbf{s}_r \ \mathbf{p}_i \ \ldots \ \mathbf{p}_n]^\top$ denote a traditional SLAM state vector. The measurements are typically corrupted by a noise process $\mathbf{n}(t)$ such that

$$d\mathbf{y}(t) \triangleq \psi dt = h(\mathbf{z})dt + \mathbf{E}(t)\mathbf{n}(t) \tag{5}$$

in continuous-time. Here, $\mathbf{n}(t)$ is a zero-mean Weiner process and $\mathbf{E}(t)$ is a measurement noise weighting matrix that can be dependent on the true state. The measurements and robot dynamics are nonlinear in the chosen coordinate system.

## III. Improved Robocentric Mapping in Polar Coordinates

The contribution of this paper is a novel robocentric algorithm for mapping and localization that takes advantage of the polar-like nature of the relative range and bearing measurements. There does not appear to be any similar (polar-like) algorithms in the SLAM or robocentric mapping literature. However, there is a long history in the bearing-only tracking literature [19], [20] of working in variants of polar coordinate systems. The motivation is that the measurements are then linear in the state components. Recall the measurements taken by the robot are in the form

$$\begin{aligned} d_i &= \sqrt{(x_i - x_r)^2 + (y_i - y_r)^2} \\ \vartheta_i &= \theta_i - \phi_r = \arctan\left(\frac{y_i - y_r}{x_i - x_r}\right) - \phi_r \end{aligned} \tag{6}$$

where the state $\mathbf{s}_r = [x_r \ y_r \ \phi_r]^\top$ of the robot and the position of the landmarks $\mathbf{p}_i = [x_i \ y_i]^\top \in \mathbb{R}^2$ are in some external (non-robocentric) coordinate system. The measurements are nonlinear in the first two components of $\mathbf{s}_r$ and in $\mathbf{p}_i$, $\forall i$.

Now define the following state variable $\mathbf{r}_i = [d_i \ \vartheta_i]^\top$ with $d_i \in (0,\infty)$ and $\vartheta_i \in [-\pi,\pi)$. The augmented state variable in this section is given by $\mathbf{z} = [\mathbf{r}_1 \ \ldots \ \mathbf{r}_n]^\top$. The measurements (6) are linear in $\mathbf{r}_i$ or more generally in $\mathbf{z} = [\mathbf{r}_1 \ \ldots \ \mathbf{r}_n]^\top$ and are given by the continuous-time measurement equation

$$d\mathbf{y}(t) \triangleq \psi dt = \mathbf{H}(\mathcal{G}(t))\mathbf{z}dt + \mathbf{E}(t)\mathbf{n}(t) \tag{7}$$

where $\mathbf{E}(t)$ is not required to be independent of $\mathbf{z}$. Here, $\mathbf{H}(\mathcal{G}(t))$ is a time-varying linear matrix which is dependent only on the set $\mathcal{G}(t)$ of currently sensed landmarks. For example, if all of the landmarks are sensed and the state variable $\mathbf{z}$ is ordered appropriately, then $\mathbf{H}$ would be the identity matrix.

Consider again a robot that obeys the unicycle model (1) in $\mathbb{R}^2 \times \mathbb{SO}(1,\mathbb{R})$. Then we can write down the following differential equation for the dynamics of $\mathbf{r}_i$,

$$\begin{aligned} \dot{d}_i &= -v_r\cos\vartheta_i \\ \dot{\vartheta}_i &= \frac{v_r}{d_i}\sin\vartheta_i - w_r \end{aligned} \tag{8}$$

which is nonlinear in $\mathbf{r}_i$. Note also that $d_i$ must be bounded away from zero. Again we (must) assume that the control inputs are corrupted by an additive noise process $v$ and $w$ such that

$$\begin{aligned} d\begin{bmatrix} d_i \\ \vartheta_i \end{bmatrix} &= \begin{bmatrix} -v_r\cos\vartheta_i \\ \frac{v_r}{d_i}\sin\vartheta_i - w_r \end{bmatrix} dt + \\ &\quad \begin{bmatrix} \sigma_v\cos\vartheta_i & 0 \\ \sigma_v\sin\vartheta_i & -\sigma_w \end{bmatrix} \begin{bmatrix} dv \\ dw \end{bmatrix} \end{aligned} \tag{9}$$

is a more accurate depiction of the relative robot and $i^{th}$ landmark dynamics. Here, $v$ and $w$ are uncorellated Weiner processes with standard deviations of $\sigma_v$ and $\sigma_w$ respectively. Note that the affect of $v$ on $\vartheta_i$ and $d_i$ is conditioned on a nonlinear function of a true state variable (in this case $\vartheta_i$).

### A. On the Observability of the Polar SLAM Problem and the Convergence of the EKF-Based Polar SLAM Algorithm

In this subsection we will examine and prove a number of results related to the observability of the considered polar-coordinate SLAM problem formulation. We will also examine and prove a number of results regarding the convergence of an EKF-like algorithm for estimating the relative polar state variable.

*1) Error Free Measurements and Dynamics:* We consider first the observability properties of the state $\mathbf{z} = [\mathbf{r}_1 \ \ldots \ \mathbf{r}_n]^\top$ with $\mathbf{r}_i = [d_i \ \vartheta_i]^\top$ evolving according to (8). We also assume error free measurements of the form

$$\psi dt = \mathbf{H}(\mathcal{G}(t))\mathbf{z}dt \tag{10}$$

such that the system and measurements are noiseless and deterministic. The following result concerns the observability of the subspace $\mathbf{r}_i = [d_i \ \vartheta_i]^\top$ for some $i \in \mathcal{V}$.

*Corollary 1:* Assume the robot-landmark dynamics and the measurements are deterministic and error free. The state $\mathbf{r}_i(s) = [d_i(s)\ \vartheta_i(s)]^\top$ for some $i \in \mathcal{V}$ and for $s \geq \tau$ or $s < \tau$ can be calculated at any time $t \geq \tau$ if and only if $\mathcal{G}(\tau) \cap \mathbf{r}_i(\tau) \neq \emptyset$ for some instant $\tau$.

The fact that Corollary 1 is true is not surprising. However, it again highlights the observable space of the SLAM problem is purely relative [9]. Hence, by considering a relative (robocentric) mapping algorithm we are not attempting to extract more information (in any finite time) from the measurements than is available [9][12].

*2) Error Free Dynamics and Noisy Measurements:* A natural extension to the above result concerns the behavior of an estimate $\widehat{\mathbf{z}}$ of $\mathbf{z}$ when the dynamics of the state $\mathbf{r}_i = [d_i\ \vartheta_i]^\top$ are error free and deterministic but the measurements

$$d\mathbf{y}(t) \triangleq \psi dt = \mathbf{H}(\mathcal{G}(t))\mathbf{z}dt + \mathbf{E}(t)\mathbf{n}(t) \tag{11}$$

are corrupted by an additive Weiner process. Naturally, the behavior of any state estimate $\widehat{\mathbf{z}}$ depends on the particular estimator and thus let us consider an estimator of the form

$$d\widehat{\mathbf{z}} = f(\widehat{\mathbf{z}}, v_r, w_r)dt + \mathbf{K}(t)\left(d\mathbf{y}(t) - \mathbf{H}(\mathcal{G}(t))\widehat{\mathbf{z}}dt\right) \tag{12}$$

where the function $f_i(\cdot)$ that captures the dynamics of the subspace $\mathbf{r}_i = [d_i\ \vartheta_i]^\top$ is given by

$$f(\widehat{\mathbf{z}}, v_r, w_r) = \begin{bmatrix} -v_r \cos\vartheta_i \\ \frac{v_r}{d_i}\sin\vartheta_i - w_r \end{bmatrix} \tag{13}$$

where $v_r$ and $w_r$ are again considered as deterministic inputs with no errors. The function $f(\cdot)$ is thus a vertical concatenation of the $f_i(\cdot)$. The gain $\mathbf{K}(t)$ is given by

$$\mathbf{K}(t) = \mathbf{P}(t)\mathbf{H}^\top(\mathcal{G}(t))\mathbf{R}^{-1}(t) \tag{14}$$

and $\mathbf{P}(t)$ is the solution to the following Riccati differential equation

$$\begin{aligned} d\mathbf{P}(t) \quad = \quad & \left[\mathbf{A}(t)\mathbf{P}(t) + \mathbf{P}(t)\mathbf{A}^\top(t) + \mathbf{Q}(t)\right]dt - \\ & \mathbf{P}(t)\mathbf{H}^\top(\mathcal{G}(t))\mathbf{R}^{-1}(t)\mathbf{H}(\mathcal{G}(t))\mathbf{P}(t) \end{aligned} \tag{15}$$

where $\mathbf{Q}$ and $\mathbf{R}$ are positive-definite tuning matrices. Note that $\mathbf{A}(t)$ is the Jacobian of $f(\cdot)$ evaluated at $\widehat{\mathbf{z}}$. The Jacobian $\mathbf{A}_i(t)$ of $f_i(\cdot)$ is given by

$$\mathbf{A}_i(t) = \begin{bmatrix} 0 & -v_r\sin\vartheta_i \\ -\frac{v_r}{d_i}\sin\vartheta_i & \frac{v_r}{d_i}\cos\vartheta_i \end{bmatrix} \tag{16}$$

and is evaluated at $\widehat{\mathbf{r}}_i$ and is dependent on $v_r$. Note the estimation error $\boldsymbol{\zeta} = \mathbf{z} - \widehat{\mathbf{z}}$ evolves according to

$$\begin{aligned} d\boldsymbol{\zeta} \quad = \quad & \left(\mathbf{A}(t) - \mathbf{K}(t)\mathbf{H}(\mathcal{G}(t))\right)\boldsymbol{\zeta}dt + \\ & \varrho(\mathbf{z}, \widehat{\mathbf{z}}, v_r, w_r)dt - \mathbf{K}(t)\mathbf{E}(t)d\mathbf{n}(t) \end{aligned} \tag{17}$$

where we have used the following Taylor expansion of $f(\cdot)$ about the estimate $\widehat{\mathbf{z}}$,

$$f(\mathbf{z}, v_r, w_r) - f(\widehat{\mathbf{z}}, v_r, w_r) = \mathbf{A}(t)(\mathbf{z} - \widehat{\mathbf{z}}) + \varrho(\mathbf{z}, \widehat{\mathbf{z}}, v_r, w_r) \tag{18}$$

where $\varrho(\mathbf{z}, \widehat{\mathbf{z}}, v_r, w_r)$ accounts for the higher order terms. Recall that $\mathbf{r}_i = [d_i\ \vartheta_i]^\top$ with $d_i \in (0, \infty)$ and $\vartheta_i \in [-\pi, \pi)$ for all $t$. Then it is clear that the following bound holds

$$\|\mathbf{A}(t)\| = \overline{a} < \infty \tag{19}$$

for all $t$ where for any time-varying matrix $\mathbf{M}(t)$ we assume the following

$$\|\mathbf{M}(t)\| = \sup\{\|\mathbf{M}(t)\| : m_{ij} \in \mathbb{R}\} \tag{20}$$

for all $t$ and for some norm $\|\cdot\|$. For the subsequent analysis, it turns out that the coordinate spatial and temporal scales will play an important role. Hence, at this point let us make the following assumptions.

*Assumption 1:* The translational velocity of the robot $v_r(t)$ is uppperbounded in any arbitrary coordinate scale such that $v_r(t) \leq \overline{v}$ for all $t$. For simplicity we also assume that $v_r(t) > 0$ for all t. Now it follows that there exists a temporal coordinate scale such that $v_r(t) \leq 1$ for all $t$.

*Assumption 2:* The relative distance between the robot and the $i^{th}$ landmark at time $t$ belongs to $d_i(t) \in (0, \infty)$ in any arbitrarily chosen coordinate scale. There exists a spatial coordinate scale such that for all $t$ we have $d_i \in [1, \infty)$.

Assumptions 1 and 2 are weak (actually notational) and can almost surely be satisfied in practice (i.e. by finding explicit spatial and temporal scales). The case of $v_r = 0$ is trivially obtained from the subsequent results. For simplicity we also assume the following.

*Assumption 3:* For all $t$ we have $\widehat{\mathbf{r}}_i(t) = [\widehat{d}_i(t)\ \widehat{\vartheta}_i(t)]^\top$ with $\widehat{d}_i \in [1, \infty)$ and $\widehat{\vartheta}_i \in [-\pi, \pi)$.

*Assumption 4:* For all $t$ we assume that the error $\zeta_{i2} = (\vartheta_i - \widehat{\vartheta}_i)$ is taken modulo $2\pi$ and $\zeta_{i2} \in [-\pi, \pi)$.

Assumption 3 calls for the state estimate components to be restricted to the assumed true global state space. For $\widehat{\vartheta}_i(t)$ this can be achieved via a trivial modular operation. Assumption 4 ensures the value of the bearing error falls within a consistent $2\pi$ interval. Finally, we make the following standard assumption regarding the design parameters

*Assumption 5:* The following $\mathbf{Q}(t) \geq \underline{q}\mathbf{I}$, $\mathbf{R}(t) \geq \underline{r}\mathbf{I}$ and $\mathbf{P}(t_0) \geq p_0\mathbf{I}$ are given for some $\underline{q}, \underline{r}, p_0 > 0$ such that $\|\mathbf{Q}(t)\| \geq \underline{q}$ and $\|\mathbf{R}(t)\| \geq \underline{r}$. Moreover, $\mathbf{Q}(t)$ and $\mathbf{R}(t)$ are chosen to be bounded by $\|\mathbf{Q}(t)\| \leq \overline{q} < \infty$ and $\|\mathbf{R}(t)\| \leq \overline{r} < \infty$ for all $t$. Also, we have $\mathbf{E}(t) \leq \overline{e} < \infty$ with $\mathbf{E}(t) \geq \underline{e}\mathbf{I}$.

We will also need the following lemma concerning the growth of $\varrho(\mathbf{z}, \widehat{\mathbf{z}}, v_r, w_r)$.

*Lemma 1: The following inequality holds*

$$\begin{aligned} \|\varrho(\mathbf{z}, \widehat{\mathbf{z}}, v_r, w_r)\| \quad = \quad & \|f(\mathbf{z}, \cdot) - f(\widehat{\mathbf{z}}, \cdot) - \mathbf{A}(t)(\mathbf{z} - \widehat{\mathbf{z}})\| \\ & \leq 2\overline{a}\|\boldsymbol{\zeta}\| \end{aligned} \tag{21}$$

*for $|\mathcal{V}| = n$ with probability 1 when Assumptions 1-5 hold.*

*Proof:* From the triangle inequality we obtain

$$\begin{aligned}\|f(\mathbf{z},v_r,w_r)-f(\widehat{\mathbf{z}},v_r,w_r)-\mathbf{A}(t)(\mathbf{z}-\widehat{\mathbf{z}})\| &\leq\\ \|f(\mathbf{z},v_r,w_r)-f(\widehat{\mathbf{z}},v_r,w_r)\|+\|-\mathbf{A}(t)\boldsymbol{\zeta}\| &\leq\\ \|f(\mathbf{z},v_r,w_r)-f(\widehat{\mathbf{z}},v_r,w_r)\|+\overline{a}\boldsymbol{\zeta} &\end{aligned} \quad (22)$$

which follows using (19). Now if $f(\cdot)$ is Lipschitz then $\|f(\mathbf{z},v_r,w_r)-f(\widehat{\mathbf{z}},v_r,w_r)\|\leq c\|\mathbf{z}-\widehat{\mathbf{z}}\|$ for some $0<c<\infty$. Actually, we know that if $\|\mathbf{A}(t)\|$ is bounded by $\overline{a}$ then $f(\cdot)$ is Lipschitz with Lipschitz coefficient $\overline{a}$. Thus, the proof is immediate. ■

Note also that $\varrho(\mathbf{z},\widehat{\mathbf{z}},v_r,w_r)=0$ when $\boldsymbol{\zeta}(t)=0$. We now consider the the propagation of the estimation error $\boldsymbol{\zeta}(t)=\mathbf{z}(t)-\widehat{\mathbf{z}}(t)$ for all $t>t_0$ given an initial estimation error $\boldsymbol{\zeta}(t_0)$ which we will assume belongs to the set

$$\boldsymbol{\zeta}(t_0)=\{\boldsymbol{\eta}\in\{[0,\infty)\times[-\pi,\pi)\}:\|\boldsymbol{\zeta}(t_0)\|\leq b\} \quad (23)$$

for some constant $b<\infty$. We assume initially that $\mathcal{G}(t)=\mathcal{V}$ for all $t>t_0$. The error propagates according to (17) with (for simplicity) $\mathbf{H}(\mathcal{G}(t))=\mathbf{I}$ for all $t$. It is common to assume a full landmark measurement vector when performing such an analysis [4], [12]. We state the following lemma regarding the error covariance.

*Lemma 2: Suppose Assumptions 1-5 hold. Then the state estimate covariance $\mathbf{P}(t)$ is bounded by*

$$0<\underline{p}\leq\mathbf{P}(t)\leq\overline{p}<\infty \quad (24)$$

*for all $t>t_0$ and where*

$$\overline{p}\triangleq\left(\|\mathbf{P}(t_0)\|+\frac{\|\mathbf{Q}(t)\|+\|\mathbf{R}(t)\|\|\boldsymbol{\Lambda}(t)\|^2}{2\kappa}\right) \quad (25)$$

*and where $\boldsymbol{\Lambda}$ is chosen such that*

$$\boldsymbol{\eta}^\top\left(\mathbf{A}(t)+\boldsymbol{\Lambda}(t)\right)\boldsymbol{\eta}\leq-\kappa\|\boldsymbol{\eta}\|^2 \quad (26)$$

*is satisfied for all $\boldsymbol{\eta}\in\mathbb{R}^2$ with $\kappa>0$.*

*Proof:* The upper bound can be obtained by considering the following time-varying linear control system

$$-\dot{\mathbf{q}}=\mathbf{A}(t)\mathbf{q}+\mathbf{u} \quad (27)$$

with a boundary $\mathbf{q}(T)=\mathbf{q}_T$ for some $0<T\leq\infty$ and with controllability Grammian

$$\mathcal{C}(t+\tau,t)=\int_t^{t+\tau}\boldsymbol{\Psi}(t+\tau,t)\boldsymbol{\Psi}^\top(t+\tau,t)dt \quad (28)$$

where $\boldsymbol{\Psi}(t+\tau,t)$ is the fundamental matrix with $\boldsymbol{\Psi}(t,t)=\mathbf{I}$. The system (27) is uniformly completely controllable since $\|\mathbf{A}(t)\|<\infty$ and $\|\boldsymbol{\Psi}(t+\tau,t)\|>\exp(-\tau\|\mathbf{A}(t)\|)$ which implies $\mathcal{C}(t+\tau,t)$ is never singular for $t_0\leq t<\tau$. Consider the following cost function

$$\mathcal{J}(t,\tau,\mathbf{q},\mathbf{u})=\mathcal{B}(t_0,\mathbf{q}(t_0))+\int_{t_0}^{T}\left(\mathbf{q}^\top\mathbf{Q}\mathbf{q}+\mathbf{u}^\top\mathbf{R}\mathbf{u}\right)dt \quad (29)$$

and value function $\mathcal{B}(t,\mathbf{q}(t))=\mathbf{q}^\top(t)\mathbf{P}(t)\mathbf{q}(t)$. Let the control input equal $\mathbf{u}(t)=\boldsymbol{\Lambda}(t)\mathbf{q}$ for some continuous bounded matrix $\boldsymbol{\Lambda}(t)$ such that $-\dot{\mathbf{q}}=\left(\mathbf{A}(t)+\boldsymbol{\Lambda}(t)\right)\mathbf{q}$. Note now that

$$\begin{aligned}\mathcal{B}(T,\mathbf{q}(T)) &= \mathbf{q}^\top(T)\mathbf{P}(T)\mathbf{q}(T)\\ &\leq \mathcal{B}(t_0,\mathbf{q}(t_0))+\\ &\quad\int_{t_0}^{T}\mathbf{q}^\top\left(\mathbf{Q}+\boldsymbol{\Lambda}^\top(t)\mathbf{R}\boldsymbol{\Lambda}(t)\right)\mathbf{q}dt\end{aligned} \quad (30)$$

Solving $-\dot{\mathbf{q}}=\left(\mathbf{A}(t)+\boldsymbol{\Lambda}(t)\right)\mathbf{q}$ for $\mathbf{q}(T)$ implies that

$$\begin{aligned}\|\mathbf{q}(T)\|^2 &= \|\mathbf{q}_T\|^2=\|\mathbf{q}(t_0)\|^2-\\ &\quad 2\int_{t_0}^{T}\mathbf{q}^\top\left(\mathbf{A}(t)+\boldsymbol{\Lambda}(t)\right)\mathbf{q}\;dt\end{aligned} \quad (31)$$

and thus (26) implies that $\|\mathbf{q}(t_0)\|^2\leq\|\mathbf{q}_T\|^2$ and $\int_{t_0}^{T}\mathbf{q}^\top\mathbf{q}\leq\frac{\|\mathbf{q}_T\|^2}{2\kappa}$. Using this with (30) leads easily to the upper-bound. ■

Note that $\|\mathbf{P}(t)\|$ is bounded above by a constant independent of the time $t>t_0$. This bound holds irrespective of whether or not the state estimation error is bounded. Part of Lemma 2 follows from a theorem given in [21]. The condition (26) calls for the system pair $\mathbf{A}(t)$ and $\mathbf{H}(\mathcal{G}(t))=\mathbf{I}$ to be uniformly detectable. In our case we know that the system is observable (which implies detectability [21], [22]). As such, a suitable matrix $\boldsymbol{\Lambda}(t)$ exists with probability one. Alternatively, an upper-bound on $\|\mathbf{P}(t)\|$ can be derived independent of (26) when the state is observable [21].

We now state a result concerning the exponential boundedness of the expected error $\mathcal{E}\{\|\boldsymbol{\zeta}(t)\|\}$ for all $t>t_0$ and the asymptotic properties of the expected estimation error.

*Theorem 1:* Consider the system (17) with an initial condition (23) and $\mathbf{H}(\mathcal{G}(t))=\mathbf{I}$. Suppose that Assumptions 1-5 hold. If $\|\mathbf{P}^{-1}(t)\mathbf{Q}(t)\mathbf{P}^{-1}(t)+\mathbf{R}^{-1}(t)\|\underline{p}>\frac{4\overline{a}\overline{p}}{\underline{p}}$ then the estimation error is bounded above with

$$\mathcal{E}\left\{\|\boldsymbol{\zeta}(t)\|^2\right\}\leq\max\left\{\frac{n\overline{p}\overline{e}^2}{2\gamma\underline{r}^2},\;\frac{\overline{p}}{\underline{p}}\|\boldsymbol{\zeta}(t_0)\|^2\right\} \quad (32)$$

where $\gamma=\|\mathbf{P}^{-1}(t)\mathbf{Q}(t)\mathbf{P}^{-1}(t)+\mathbf{R}^{-1}(t)\|\underline{p}-4\overline{a}\overline{p}/\underline{p}$ and the error $\mathcal{E}\left\{\|\boldsymbol{\zeta}(t)\|^2\right\}$ as $t\to\infty$ is bounded by $\frac{n\overline{p}\overline{e}^2}{2\gamma\underline{r}^2}$.

*Proof:* The error system (17) can be thought of as a linear system with a nonlinear perturbation being driven by a zero-mean Weiner process. Let $\mathcal{B}(t,\boldsymbol{\zeta}(t))=\boldsymbol{\zeta}^\top(t)\mathbf{P}^{-1}(t)\boldsymbol{\zeta}(t)>0$ and note that

$$\begin{aligned}d\mathcal{B} &= \left[\frac{\partial\mathcal{B}}{\partial t}+\frac{\partial\mathcal{B}}{\partial\boldsymbol{\zeta}}\left(\mathbf{A}(t)-\mathbf{K}(t)\right)\boldsymbol{\zeta}\right]dt+\\ &\quad\frac{\partial\mathcal{B}}{\partial\boldsymbol{\zeta}}\varrho(\mathbf{z},\widehat{\mathbf{z}},v_r,w_r)dt+\\ &\quad\frac{1}{2}\mathrm{tr}\left(\mathrm{hess}(\mathcal{B})\mathbf{K}(t)\mathbf{E}(t)\mathbf{E}^\top(t)\mathbf{K}^\top(t)\right)dt-\\ &\quad\frac{\partial\mathcal{B}}{\partial\boldsymbol{\zeta}}\mathbf{K}(t)\mathbf{E}(t)d\mathbf{n}\\ d\mathcal{B} &= \left[\frac{\partial\mathcal{B}}{\partial t}+\mathcal{L}\mathcal{B}\right]dt-\frac{\partial\mathcal{B}}{\partial\boldsymbol{\zeta}}\mathbf{K}(t)\mathbf{E}(t)d\mathbf{n}\end{aligned} \quad (33)$$

using Ito's differential formula and where $\mathcal{L}$ is the Kolmogorov backward operator, $\mathrm{hess}(\cdot)$ denotes the Hessian

operator and $\mathrm{tr}(\cdot)$ denotes the matrix trace. Evaluating the terms and re-arranging leads to

$$\begin{aligned} d\mathcal{B} &= \left[\boldsymbol{\zeta}^\top \left[\mathbf{P}^{-1}(t)\mathbf{Q}(t)\mathbf{P}^{-1}(t) + \mathbf{R}^{-1}(t)\right]\boldsymbol{\zeta}\right] dt + \\ &\quad 2\boldsymbol{\zeta}^\top \mathbf{P}^{-1}(t)\varrho(\mathbf{z}, \widehat{\mathbf{z}}, v_r, w_r)dt + \\ &\quad \frac{1}{2}\mathrm{tr}\left(\mathbf{R}^{-1}(t)\mathbf{E}(t)\mathbf{E}(t)\mathbf{R}^{-\top}(t)\mathbf{P}^\top(t)\right) dt - \\ &\quad 2\boldsymbol{\zeta}^\top \mathbf{R}^{-1}(t)d\mathbf{n} \\ &\leq \left[-\alpha\|\boldsymbol{\zeta}\|^2 + \frac{4\overline{a}}{\underline{p}}\|\boldsymbol{\zeta}\|^2 + \frac{n\overline{p}\overline{e}^2}{2\underline{r}^2}\right] dt - \\ &\quad 2\boldsymbol{\zeta}^\top \mathbf{R}^{-1}(t)d\mathbf{n} \end{aligned} \tag{34}$$

where we have explicitly employed Lemma 1 and Lemma 2 and where

$$\alpha = \|\mathbf{P}^{-1}(t)\mathbf{Q}(t)\mathbf{P}^{-1}(t) + \mathbf{R}^{-1}(t)\| \tag{35}$$

Clearly we have $\overline{p}^{-1}\|\boldsymbol{\zeta}\|^2 \leq \mathcal{B}(t, \boldsymbol{\zeta}(t)) \leq \underline{p}^{-1}\|\boldsymbol{\zeta}\|^2$ such that some simple algebra implies that

$$\begin{aligned} d\mathcal{B} &\leq -\left(\alpha\underline{p} - \frac{4\overline{a}\overline{p}}{\underline{p}}\right)\mathcal{B}dt + \frac{n\overline{p}\overline{e}^2}{2\underline{r}^2}dt - \\ &\quad 2\boldsymbol{\zeta}^\top \mathbf{R}^{-1}(t)d\mathbf{n} \\ \mathcal{B} &\leq \mathcal{B}(t_0, \boldsymbol{\zeta}(t_0)) - \\ &\quad \int_{t_0}^{t}\left(\alpha\underline{p} - \frac{4\overline{a}\overline{p}}{\underline{p}}\right)\mathcal{B}(\tau, \boldsymbol{\zeta}(\tau))d\tau + \\ &\quad \frac{n\overline{p}\overline{e}^2}{2\underline{r}^2}\int_{t_0}^{t} d\tau - 2\int_{t_0}^{t}\boldsymbol{\zeta}^\top(\tau)\mathbf{R}^{-1}(\tau)d\mathbf{n}(\tau) \end{aligned} \tag{36}$$

From the Bellman-Gromwall lemma [23] we have

$$\begin{aligned} \mathcal{B}(t, \boldsymbol{\zeta}(t)) &\leq \mathcal{B}(t_0, \boldsymbol{\zeta}(t_0))\exp\left(-\gamma(t - t_0)\right) + \\ &\quad \frac{n\overline{p}\overline{e}^2}{2\gamma\underline{r}^2}\left(1 - \exp\left(-\gamma(t - t_0)\right)\right) - \\ &\quad 2\int_{t_0}^{t}\boldsymbol{\zeta}^\top(\tau)\mathbf{R}^{-1}(\tau)d\mathbf{n}(\tau) \end{aligned} \tag{37}$$

where

$$\gamma = \left(\alpha\underline{p} - 4\overline{a}\overline{p}/\underline{p}\right) \tag{38}$$

with $\gamma > 0$ if and only if $\alpha\underline{p} > \frac{4\overline{a}\overline{p}}{\underline{p}}$. Taking the expectation $\mathcal{E}\{\cdot\}$ of both sides of (37) gives

$$\begin{aligned} \mathcal{E}\left\{\mathcal{B}(t, \boldsymbol{\zeta}(t))\right\} &\leq \mathcal{B}(t_0, \boldsymbol{\zeta}(t_0))\exp\left(-\gamma(t - t_0)\right) + \\ &\quad \frac{n\overline{p}\overline{e}^2}{2\gamma\underline{r}^2}\left(1 - \exp\left(-\gamma(t - t_0)\right)\right) \end{aligned} \tag{39}$$

and thus

$$\begin{aligned} \mathcal{E}\left\{\|\boldsymbol{\zeta}(t)\|^2\right\} &\leq \frac{\overline{p}}{\underline{p}}\|\boldsymbol{\zeta}(t_0)\|^2\exp\left(-\gamma(t - t_0)\right) + \\ &\quad \frac{n\overline{p}\overline{e}^2}{2\gamma\underline{r}^2}\left(1 - \exp\left(-\gamma(t - t_0)\right)\right) \end{aligned} \tag{40}$$

We then easily find that

$$\mathcal{E}\left\{\|\boldsymbol{\zeta}(t)\|^2\right\} \leq \max\left\{\frac{n\overline{p}\overline{e}^2}{2\gamma\underline{r}^2},\ \frac{\overline{p}}{\underline{p}}\|\boldsymbol{\zeta}(t_0)\|^2\right\} \tag{41}$$

for all $t$ if $\gamma > 0$ and the error $\mathcal{E}\left\{\|\boldsymbol{\zeta}(t)\|^2\right\}$ as $t \to \infty$ is bounded by $\frac{n\overline{p}\overline{e}^2}{2\gamma\underline{r}^2}$. This completes the proof. ■

Importantly, we have shown under what conditions an EKF-like algorithm will yield an exponentially bounded and converging mean-square estimation error. The condition $\gamma > 0$, which guarantees the expected error converges, is independent of $\overline{e}$. We have also given a method of estimating the asymptotic mean-square error. The asymptotic mean-square estimation error is dependent on the specific robot trajectory but is upper-bounded by $\frac{n\overline{p}\overline{e}^2}{2\gamma\underline{r}^2}$. Theorem 1 is a significant contribution to the problem of robocentric mapping and is a fundamental result. It is important to note again that the algorithm considered in this paper is based on nothing more than an EKF-like architecture and a coordinate transform; see [20], [24], [25] for other EKF stability results.

*3) Noisy Robot-Landmark Dynamics and Noisy Measurements:* We now consider the case where process noise is present and where (for simplicity) $\mathbf{H}(\mathcal{G}(t)) = \mathbf{I}$ for all $t$. We assume an EKF-like algorithm of the form (12) with Assumptions 1-5 holding. The error $\boldsymbol{\zeta} = \mathbf{z} - \widehat{\mathbf{z}}$ obeys

$$\begin{aligned} d\boldsymbol{\zeta} &= \left[\left(\mathbf{A}(t) - \mathbf{K}(t)\right)\boldsymbol{\zeta} + \varrho(\mathbf{z}, \widehat{\mathbf{z}}, v_r, w_r)\right] dt + \\ &\quad \mathbf{G}(t)\begin{bmatrix} dv \\ dw \end{bmatrix} - \mathbf{K}(t)\mathbf{E}(t)d\mathbf{n}(t) \\ d\boldsymbol{\zeta} &= \left[\left(\mathbf{A}(t) - \mathbf{K}(t)\right)\boldsymbol{\zeta} + \varrho(\mathbf{z}, \widehat{\mathbf{z}}, v_r, w_r)\right] dt + \\ &\quad \left[\mathbf{G}(t)\ -\mathbf{K}(t)\mathbf{E}(t)\right]\begin{bmatrix} \begin{bmatrix} dv \\ dw \end{bmatrix} \\ d\mathbf{n}(t) \end{bmatrix} \end{aligned} \tag{42}$$

where $\mathbf{G}_i(t)$ is given by

$$\|\mathbf{G}_i(t)\| = \left\|\begin{bmatrix} \sigma_v\cos\vartheta_i & 0 \\ \sigma_v\sin\vartheta_i & -\sigma_w \end{bmatrix}\right\| = \overline{g} < \infty \tag{43}$$

and $\mathbf{G}(t) = \left[\mathbf{G}_1(t)\ \ldots\ \mathbf{G}_n(t)\right]^\top$. Moreover, Lemma 1 and Lemma 2 still apply since they depend only on the validity of Assumptions 1-5. Now we are in a position to prove the main result concerning the exponential boundedness of the expected estimation error for all $t > t_0$.

*Theorem 2:* Consider the system (42) with an initial condition (23) and $\mathbf{H}(\mathcal{G}(t)) = \mathbf{I}$. Suppose that Assumptions 1-5 hold. If $\|\mathbf{P}^{-1}(t)\mathbf{Q}(t)\mathbf{P}^{-1}(t) + \mathbf{R}^{-1}(t)\|\underline{p} > \frac{4\overline{a}\overline{p}}{\underline{p}}$ then the estimation error is bounded above with

$$\mathcal{E}\left\{\|\boldsymbol{\zeta}(t)\|^2\right\} \leq \max\left\{\frac{n(\underline{r}^2\overline{g}^2 + \overline{p}\underline{p}\overline{e}^2)}{2\gamma\underline{r}^2\underline{p}},\ \frac{\overline{p}}{\underline{p}}\|\boldsymbol{\zeta}(t_0)\|^2\right\} \tag{44}$$

where $\gamma = \|\mathbf{P}^{-1}(t)\mathbf{Q}(t)\mathbf{P}^{-1}(t) + \mathbf{R}^{-1}(t)\|\underline{p} - 4\overline{a}\overline{p}/\underline{p}$ and the error $\mathcal{E}\left\{\|\boldsymbol{\zeta}(t)\|^2\right\}$ as $t \to \infty$ is bounded by $\frac{n(\underline{r}^2\overline{g}^2 + \overline{p}\underline{p}\overline{e}^2)}{2\gamma\underline{r}^2\underline{p}}$.

*Proof:* The proof is similar to the proof of Theorem 1. We will omit most of the details as a consequence. Let $\mathcal{B}(t, \boldsymbol{\zeta}(t)) = \boldsymbol{\zeta}^\top(t)\mathbf{P}^{-1}(t)\boldsymbol{\zeta}(t) > 0$ and note that

$$\begin{aligned} d\mathcal{B} &= \left[\frac{\partial\mathcal{B}}{\partial t} + \frac{\partial\mathcal{B}}{\partial\boldsymbol{\zeta}}\left(\mathbf{A}(t) - \mathbf{K}(t)\right)\boldsymbol{\zeta} + \frac{\partial\mathcal{B}}{\partial\boldsymbol{\zeta}}\varrho(\mathbf{z}, \widehat{\mathbf{z}}, \cdot)\right] dt + \\ &\quad dt + \\ &\quad \frac{1}{2}\mathrm{tr}\left(\mathrm{hess}(\mathcal{B})\boldsymbol{\Xi}(t)\boldsymbol{\Xi}^\top(t)\right) dt - \\ &\quad \frac{\partial\mathcal{B}}{\partial\boldsymbol{\zeta}}\boldsymbol{\Xi}(t)\begin{bmatrix} \begin{bmatrix} dv \\ dw \end{bmatrix} \\ d\mathbf{n}(t) \end{bmatrix} \end{aligned} \tag{45}$$

where

$$\boldsymbol{\Xi}(t) = [\mathbf{G}(t) \ -\mathbf{K}(t)\mathbf{E}(t)] \tag{46}$$

and where we have employed Itos differential formula. Evaluating the terms and re-arranging leads to

$$\begin{aligned} d\mathcal{B} &= \left[\boldsymbol{\zeta}^\top \left[\mathbf{P}^{-1}(t)\mathbf{Q}(t)\mathbf{P}^{-1}(t) + \mathbf{R}^{-1}(t)\right]\boldsymbol{\zeta}\right] dt + \\ &\quad 2\boldsymbol{\zeta}^\top \mathbf{P}^{-1}(t)\varrho(\mathbf{z}, \widehat{\mathbf{z}}, v_r, w_r)dt + \\ &\quad \frac{1}{2}\mathrm{tr}\left(\mathbf{P}^{-1}(t)\mathbf{G}(t)\mathbf{G}^\top(t)\right) dt + \\ &\quad \frac{1}{2}\mathrm{tr}\left(\mathbf{P}^{-1}(t)\mathbf{K}(t)\mathbf{K}^\top(t)\right) dt - \\ &\quad 2\boldsymbol{\zeta}^\top \mathbf{P}^{-1}(t)\boldsymbol{\Xi}(t)\begin{bmatrix}\begin{bmatrix} dv \\ dw \end{bmatrix} \\ d\mathbf{n}(t)\end{bmatrix} \\ &\leq \left[-\alpha\|\boldsymbol{\zeta}\|^2 + \frac{4\overline{a}}{\underline{p}}\|\boldsymbol{\zeta}\|^2 + \frac{n(\underline{r}^2\overline{g}^2 + \overline{p}\underline{p}\overline{e}^2)}{2\gamma\underline{r}^2\underline{p}}\right] dt - \\ &\quad 2\boldsymbol{\zeta}^\top \mathbf{P}^{-1}(t)\boldsymbol{\Xi}(t)\begin{bmatrix}\begin{bmatrix} dv \\ dw \end{bmatrix} \\ d\mathbf{n}(t)\end{bmatrix} \end{aligned} \tag{47}$$

where we have explicitly employed Lemma 1 and Lemma 2 and where

$$\alpha = \|\mathbf{P}^{-1}(t)\mathbf{Q}(t)\mathbf{P}^{-1}(t) + \mathbf{R}^{-1}(t)\| \tag{48}$$

Now noting that $\overline{p}^{-1}\|\boldsymbol{\zeta}\|^2 \leq \mathcal{B}(t, \boldsymbol{\zeta}(t)) \leq \underline{p}^{-1}\|\boldsymbol{\zeta}\|^2$ and using the Bellman-Gromwall lemma [23] we come to

$$\begin{aligned} \mathcal{B}(t, \boldsymbol{\zeta}(t)) &\leq \mathcal{B}(t_0, \boldsymbol{\zeta}(t_0))\exp\left(-\gamma(t - t_0)\right) - \\ &\quad 2\int_{t_0}^{t} \boldsymbol{\zeta}^\top(\tau)\mathbf{P}^{-1}(\tau)\boldsymbol{\Xi}(\tau)\begin{bmatrix}\begin{bmatrix} dv(\tau) \\ dw(\tau) \end{bmatrix} \\ d\mathbf{n}(\tau)\end{bmatrix} + \\ &\quad \frac{n(\underline{r}^2\overline{g}^2 + \overline{p}\underline{p}\overline{e}^2)}{2\gamma\underline{r}^2\underline{p}} - \\ &\quad \frac{n(\underline{r}^2\overline{g}^2 + \overline{p}\underline{p}\overline{e}^2)}{2\gamma\underline{r}^2\underline{p}}\exp\left(-\gamma(t - t_0)\right) \end{aligned} \tag{49}$$

where

$$\gamma = \left(\alpha\underline{p} - 4\frac{\overline{a}\overline{p}}{\underline{p}}\right) \tag{50}$$

with $\gamma > 0$ if and only if $\alpha\underline{p} > \frac{4\overline{a}\overline{p}}{\underline{p}}$. Taking the expectation of (49) and proceeding as in the proof of Theorem 1 gives

$$\mathcal{E}\left\{\|\boldsymbol{\zeta}(t)\|^2\right\} \leq \max\left\{\frac{n(\underline{r}^2\overline{g}^2 + \overline{p}\underline{p}\overline{e}^2)}{2\gamma\underline{r}^2\underline{p}}, \ \frac{\overline{p}}{\underline{p}}\|\boldsymbol{\zeta}(t_0)\|^2\right\} \tag{51}$$

for all $t$ if $\gamma > 0$. The error $\mathcal{E}\left\{\|\boldsymbol{\zeta}(t)\|^2\right\}$ as $t \to \infty$ is bounded by $\frac{n(\underline{r}^2\overline{g}^2 + \overline{p}\underline{p}\overline{e}^2)}{2\gamma\underline{r}^2\underline{p}}$. This completes the proof. ■

Again we have a fundamental result concerning the exponential boundedness and convergence of the expected estimation error. Note that the steady state expected mean square error bound is larger when process noise is present (as expected).

## IV. Numerical Simulations

The algorithm presented in this paper is now illustrated via simulation. The examples we consider involve a single mobile robot and a rectangular configuration of 40 landmarks. The scenario is illustrated graphically in Figure 1.

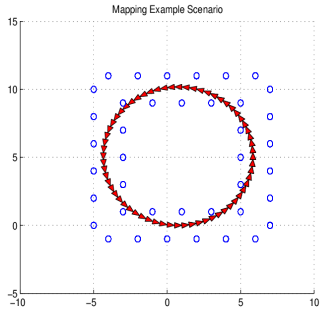



Fig. 1. The example scenario considered in this paper consists of 40 landmarks and a single mobile robot. The true robot trajectory is illustrated by the sequence of arrow heads (and starts at the origin).

The true robot velocity has a magnitude $v_r = 0.5$. The true angular velocity of the robot has magnitude $w_r = \frac{\pi}{32}$. The matrix $\mathbf{Q}(t) = \mathbf{G}(t)\mathbf{G}^\top(t)$ is evaluated at the estimated target state but uses the true values of $\sigma_v$ and $\sigma_w$. The matrix $\mathbf{R}$ is a constant matrix representing the true covariance of the measurement vector. The initial landmark positions are equal to the first noisy measurements and the associated initial covariance matrix is equal to $\mathbf{R}$. This initialization method is a very convenient side benefit of our approach.

### A. Example 1

Firstly we consider the case in which the robot senses the entire set $\mathcal{V}$ of landmarks for all $t$. This is of course not entirely practical but is the precise condition under which the analysis of this paper pertains to (and is a common assumption made when analyzing the convergence of mapping and/or SLAM algorithms). The velocity error has standard deviation magnitude of $\sigma_v = 0.05$. The angular acceleration error has a standard deviation magnitude of $\sigma_w = 0.0175$. The bearing and range noise are assumed to be independent of the state with standard deviation magnitudes of $0.035$ and $0.5$ respectively. The RMS state estimation error for 10000 simulation runs is shown in Figure 2.

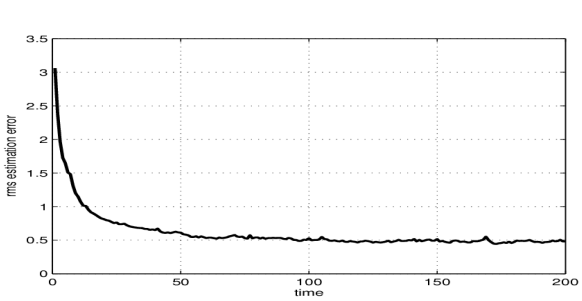



Fig. 2. The RMS state estimation error for the complete relative (robocentric) range and bearing state estimate for example 1.

It is clear that the error is bounded above and converging to a steady value. Note that the error vector consists of bearing and range errors (as we are working in polar coordinates). However, each state component is bounded and convergent. The error covariance also converges and we plot the average maximum singular value of $\mathbf{P}(t)$ in Figure 3.

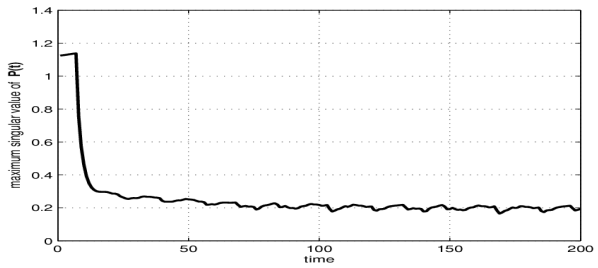



Fig. 3. The average maximum singular value of $\mathbf{P}(t)$ for example 1.

It is clear that the maximum singular value of the covariance matrix $\mathbf{P}(t)$ converges very fast to a steady state value. The average value of $\alpha$ and $\|\mathbf{A}(t)\|$ is shown in Figure 4.

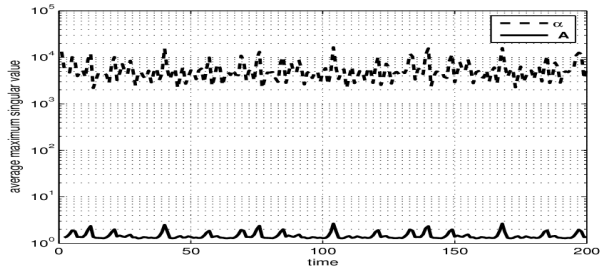



Fig. 4. The mean value of $\alpha$ and $\|\mathbf{A}(t)\|$ for example 1.

From Figure 3 and Figure 4 we can verify that the simulation results agree with the theoretical results provided in this paper.

### B. Example 2

The example considered here is identical to the previous example except for the values of the noise variances. Here we increase the values such that $\sigma_v = 0.1$ and $\sigma_w = 0.035$. The bearing and range noise standard deviations are $0.175$ and $1.5$ respectively. These are large error statistics. The RMS state error value for 10000 simulation runs is shown in Figure 5.

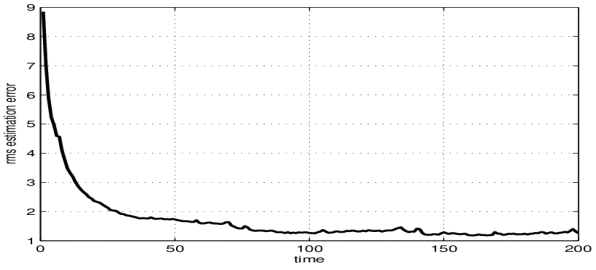



Fig. 5. The RMS state estimation error for the complete relative (robocentric) range and bearing state estimate for example 2.

The RMS error is bounded above and converging. The value to which the error is converging is also greater than that value indicated in Figure 2 for example 1 (as expected) and it takes slightly longer for the error to reach a steady-value. The covariance matrix $\mathbf{P}(t)$ or more specifically $\|\mathbf{P}(t)\|$ also converges to a steady state value as expected. It can similarly be shown (as was the case in example 1) that the condition $\gamma > 0$ is satisfied in this simulation example (given relatively large error statistics).

A significant advantage exhibited by the algorithm considered in this paper is the coordinate transformation that subsequently permits linear measurements. This can considerably improve the performance of the EKF as shown here and in the bearing-only tracking literature [19], [20].

### C. Example 3

Finally, we consider the same noise parameters and simulation scenario as examined in example 1 but we restrict the sensing domain of the robot such that it can only sense a subset $\mathcal{G} \subseteq \mathcal{V}$ of landmarks at each time $t$. Specifically, the robot can sense a landmark $i$ at time $s$ if and only if $\vartheta_i(s) \in (-\pi/2, \pi/2)$ and $d_i(s) \in (0, 5)$. We assume perfect data-association capabilities. The duration of each simulation run is increased to 400 seconds. We plot the RMS state error value over 10000 simulation runs in Figure 6.

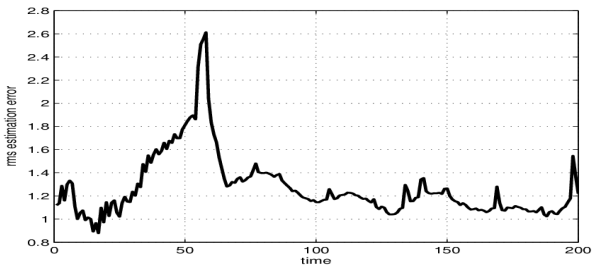



Fig. 6. The RMS state estimation error for the complete relative ( robocentric) range and bearing state estimate for example 3.

The robot completes one cycle and closes-the-loop in just over 50 seconds. The error is increasing as the robot moves from its initial position at the origin around the first loop. When the robot completes one loop we see a notable (and sudden) decrease in the error which then converges to a reasonably stable value. We plot the average maximum singular value of $\mathbf{P}(t)$ in Figure 7.

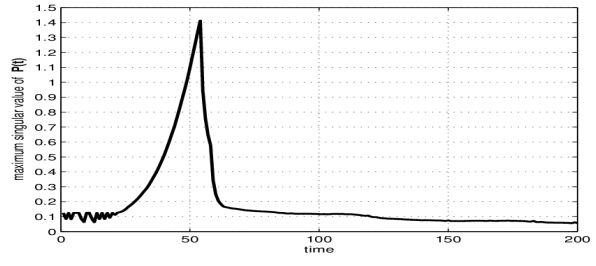



Fig. 7. The average maximum singular value of $\mathbf{P}(t)$ for example 3.

A similar situation is observed with the singular values of the covariance matrix $\mathbf{P}(t)$. During the first loop the uncertainty is increasing and following the loop-closure the uncertainty decreases dramatically.

## V. Discussion

Consider the state variable $\mathbf{z} = [\mathbf{s}_r \ \mathbf{r}_1 \ \dots \ \mathbf{r}_n]^\top$ and the corresponding EKF algorithm used to estimate $\mathbf{z}$. The state estimation of the subspace $[\mathbf{r}_1 \ \dots \ \mathbf{r}_n]^\top$ would obey the analytical results derived in this paper while the state estimation of $[\mathbf{s}_r]$ would depend on the nonlinear measurements and is not covered explicitly in this paper. Actually, $[\mathbf{s}_r]$ represents an unobservable subspace of $\mathbf{z} = [\mathbf{s}_r \ \mathbf{r}_1 \ \dots \ \mathbf{r}_n]^\top$. Thus it is possible to directly relate the robocentric mapping algorithm developed in this paper to the general global SLAM problem by simply further augmenting the state variable as $\mathbf{z} = [\mathbf{s}_r \ \mathbf{r}_1 \ \dots \ \mathbf{r}_n]^\top$ and modifying (in an obvious way) certain properties of the EKF algorithm. The result would be an algorithm for an unobservable state variable. The observable state space $[\mathbf{r}_1 \ \dots \ \mathbf{r}_n]^\top$ is the robocentric output and the estimation error associated with the space $[\mathbf{r}_1 \ \dots \ \mathbf{r}_n]^\top$ would obey the results developed in this paper. Of course the entire state $\mathbf{z}$ might diverge and there is no guarantee that the entire $\mathbf{P}(t)$ matrix is bounded (this would depend non-trivially on the robot trajectory and initialization).

Further experimental and simulation results will appear in an extended version of this paper. The results given here were simplified in an attempt to highlight the main convergence properties of the filter. A comparison of the proposed algorithm with that of the traditional formulation of EKF-SLAM is warranted along with an analysis of the *degree-of-nonlinearity* of the converted dynamic model.

## VI. Concluding Remarks

Robocentric mapping provides an attractive and tractable solution to many problems in robotics. The approach introduced in this paper is based on nothing more than an extended Kalman filter (EKF) and a very advantageous coordinate transform. The novelty of this transformation is that it leads to a linear measurement equation, i.e. it removes significant nonlinearities associated with the measurements. The standard unicycle model is given in polar coordinates and relative to each landmark position. Hence, the robocentric mapping problem given range and bearing measurements is formulated in (arguably) its most natural form. To justify the application of the EKF we then analyzed the finite-time and the asymptotic convergence properties of the error. We showed how the performance of the EKF estimation error can be related to the design parameters and the noise properties.

## References


[1] R. Smith, M. Self, and P. Cheeseman. Estimating uncertain spatial relationships in robotics. *Autonomous Robot Vehicles*, pages 167–193, Springer-Verlag 1990.

[2] H. F. Durrant-Whyte. Uncertain geometry in robotics. *IEEE Transactions on Robotics and Automation*, 4:23–31, 1988.

[3] S. Thrun, D. Fox, and W. Burgard. A probabilistic approach to concurrent mapping and localization for mobile robots. *Mach. Learning Autonom. Robots*, 31:29–53, 1998.

[4] M.W.M.G. Dissanayake, P. Newman, S. Clark, H.F. Durrant-Whyte, and M. Csorba. A solution to the simultaneous localization and map building (SLAM) problem. *IEEE Transactions on Robotics and Automation*, 17(3):229–241, June 2001.

[5] Estrada C., Neira J., and Tardos J.D. Hierarchical SLAM: Real-time accurate mapping of large environments. *IEEE Transactions on Robotics*, 21(4):588–596, August 2005.

[6] Walter M.R., Eustice R.M., and Leonard J.J. Exactly sparse extended information filters for feature-based SLAM. *International Journal of Robotics Research*, 26(4):335–359, April 2007.

[7] H. Durrant-Whyte and T. Bailey. Simultaneous localization and mapping (SLAM): Part I: The essential algorithms. *IEEE Robotics & Automation Magazine*, 13(2):99–108, 2006.

[8] T. Bailey and H. Durrant-Whyte. Simultaneous localization and mapping (SLAM): Part II: State of the art. *IEEE Robotics & Automation Magazine*, 13(3):108–117, 2006.

[9] G.P. Huang, A.I. Mourikis, and S.I. Roumeliotis. Analysis and improvement of the consistency of extended Kalman filter based SLAM. In *Proceedings of the 2008 International Conference on Robotics and Automation (ICRA)*, pages 473–479, May 2008.

[10] S.J. Julier and J.K. Uhlmann. A counter example to the theory of simultaneous localization and map building. In *Proceedings of the 2001 International Conference on Robotics and Automation (ICRA)*, pages 4238–4243, May 2001.

[11] T. Bailey, J. Nieto, J. Guivant, M. Stevens, and E. Nebot. Consistency of the EKF-SLAM algorithm. In *Proceedings of the 2006 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, page 35623568, October 2006.

[12] S. Huang and G. Dissanayake. Convergence and consistency analysis for extended kalman filter based SLAM. *IEEE Transactions on Robotics*, 23(5):1036–1049, October 2007.

[13] Castellanos J.A., Martinez-Cantin R., Tardos J.D., and Neira J. Robocentric map joining: Improving the consistency of EKF-SLAM. *Robotics and Autonomous Systems*, 55(1):21–29, January 2007.

[14] F. Bourgault, A.A. Makarenko, S.B. Williams, B. Grocholsky, , and H.F. Durrant-Whyte. Information based adaptive robotic exploration. In *Proceedings of the 2002 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2002.

[15] B. Grocholsky, J. Keller, V. Kumar, and G. Pappas. Cooperative air and ground surveillance. *IEEE Robotics & Automation Magazine*, 13(3):16–25, September 2006.

[16] A.N. Bishop, B. Fidan, B.D.O. Anderson, K. Dogancay, and P.N. Pathirana. Optimality analysis of sensor-target geometries in passive localization: Part 1 - Bearing-only localization. In *Proceedings of the 3rd International Conference on Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP)*, December 2007.

[17] A.N. Bishop, B. Fidan, B.D.O. Anderson, P.N. Pathirana, and K. Dogancay. Optimality analysis of sensor-target geometries in passive localization: Part 2 - Time-of-arrival based localization. In *Proceedings of the 3rd International Conference on Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP)*, December 2007.

[18] A.N. Bishop, B. Fidan, K. Dogancay, B.D.O. Anderson, and P.N. Pathirana. Exploiting geometry for improved hybrid AOA/TDOA based localization. *Signal Processing*, 88(7):17751791, July 2008.

[19] V. Aidala and S. Hammel. Utilization of modified polar coordinates for bearings-only tracking. *IEEE Transactions on Automatic Control*, 28(3):283294, March 1983.

[20] T. Song and J. Speyer. A stochastic analysis of a modified gain extended Kalman filter with applications to estimation with bearings only measurements. *IEEE Transactions on Automatic Control*, 30(10):940949, October 1985.

[21] J.S. Baras, A. Bensoussan, and M.R. James. Dynamic observers as asymptotic limits of recursive filters: Special cases. *SIAM Journal on Applied Mathematics*, 48(5):11471158, October 1988.

[22] B.D.O. Anderson and J.B. Moore. *Optimal Control: Linear Quadratic Methods*. Prentice-Hall, Englewood Cliffs, N.J., 1989.

[23] S. Sastry. *Nonlinear Systems: Analysis, Stability and Control*. Springer-Verlag, New York, N.Y., 1999.

[24] K. Reif, S. Gunther, E. Yaz, , and R. Unbehauen. Stochastic stability of the discrete-time extended Kalman filter. *IEEE Transactions on Automatic Control*, 44(4):714728, April 1999.

[25] K. Reif, S. Gunther, E. Yaz, , and R. Unbehauen. Stochastic stability of the continuous-time extended Kalman filter. *IEE Proceedings - Control Theory and Applications*, 147(1):4552, January 2000.

# Robocentric Mapping and Localization in Modified Spherical Coordinates with Bearing Measurements

Anders Boberg, Adrian N. Bishop and Patric Jensfelt

*Centre for Autonomous Systems (CAS), Royal Institute of Technology (KTH)*
aboberg@kth.se, adrian.bishop@ieee.org, patric@kth.se

***Abstract*—In this paper, a new approach to robotic mapping is presented that uses modified spherical coordinates in a robot-centered reference frame and a bearing-only measurement model. The algorithm provided in this paper permits robust delay-free state initialization and is computationally more efficient than the current standard in bearing-only (delay-free initialized) simultaneous localization and mapping (SLAM). Importantly, we provide a detailed nonlinear observability analysis which shows the system is generally observable. We also analyze the error convergence of the filter using stochastic stability analysis. We provide an explicit bound on the asymptotic mean state estimation error. A comparison of the performance of this filter is also made against a standard world-centric SLAM algorithm in a simulated environment.**

## I. Introduction

Simultaneous localization and mapping (SLAM) is a well researched problem within robotics. Many implementations and scenario variations exist using a variety of different filters [1]–[4]. However, it is surprising that within the SLAM literature, there is relatively little research on the use of, and subsequent performance surrounding, different coordinate systems or on the analysis of the filter error convergence. In the closely related field of target tracking, research has shown that coordinate transforms that linearize the measurement model may improve error convergence [5], [6]. Indeed, in traditional target tracking [5] the system dynamic model is often originally linear in Cartesian coordinates. However, by changing coordinates in order to derive an analytically linear measurement model we typically sacrifice this linearity of the system model. Nevertheless, overall estimation performance is often improved as discussed in [5], [6]. In robotic mapping and localization algorithms we typically start with a nonlinear system model in any case. Moreover, in the typical world-centric SLAM formulation we start with an unobservable [7], [8] nonlinear (in both system and measurements) state estimation problem.

The unobservability of the world-centric SLAM problem [7], [8] suggests that a robot-centric formulation may be more appropriate. Moreover, estimator inconsistencies caused by accumulated linearization errors [9]–[11] are exasperated in world-centric coordinates, particularly for extended Kalman filter-like (EKF) algorithms. In [12] the concept of robocentric mapping is introduced and shown to better deal with linearization errors than the traditional SLAM formulation.

This work was supported by the Swedish Foundation for Strategic Research (SSF) through CAS and also via the EU FP7 project 'CogX'.

One contribution of this paper is an algorithm for robocentric bearings-only SLAM which uses a modified spherical coordinate system. The problem of bearing-only SLAM is of interest since many sensors are capable only of providing the bearing of target-points-of-interest. For example, single camera, vision-based, measurement systems provide only the bearings to particular points in three-dimensional space [13]. Similarly, passive sensing technology often provides only target bearing information. By building a map in a relative spherical-like framework, we eliminate the nonlinearities associated with the measurement equation. Moreover, we eliminate the problems associated with the unobservable states [7] and the inconsistencies caused by the EKF linearizations (which alter the unobservable subspace [7]).

Another important contribution of this paper is the inclusion of a rigorous observability analysis. We show that in general, our robocentric system state is observable, i.e. the relative location of the landmarks are observable, given only relative bearing measurements. We go further than this and provide conditions under which the state estimation error of an EKF-like algorithm is bounded. The convergence analysis in this paper is actually conservative, with the particular asymptotic properties of the mean estimation error dependent on the exact robot trajectory; e.g. see [14], [15]. The analysis in this paper is provided in order to justify the modified spherical coordinates considered, and the wide application of the EKF in mapping (and, in particular, in mapping in this coordinate framework).

The work in this paper differs from that in the target tracking literature since we consider a nonlinear robot dynamic model. We then rigorously analyze the observability and convergence of an EKF-like algorithm given the particular nonlinear dynamics, bearing-only measurements and the modified coordinate system. We differ from related work in the robot mapping and localization literature by introducing a new coordinate system, within which a number of distinct advantages are shown to exist. We also differ from existing robotic mapping papers by introducing a rigorous convergence and observability analysis for the estimation problem. This analysis will be of interest to roboticists employing similarly structured algorithms.

## II. Preliminaries

We assume a robot moving on a planar surface according to the unicycle motion model. The robot state is described by the vector $\mathbf{x}_r = [x_r \ y_r \ z_r \ \phi_r]^{\mathrm{T}}$. The robot is steered using

the inputs $v_r$ and $\omega_r$, which are the translational and angular velocities. The robot's motion is described by the following system of nonlinear equations

$$\begin{array}{rcl} \dot{x}_r &=& v_r \cos\phi_r \\ \dot{y}_r &=& v_r \sin\phi_r \\ \dot{z}_r &=& 0 \\ \dot{\phi}_r &=& w_r \end{array} \tag{1}$$

These inputs are disturbed by the noise components $v_n$ and $\omega_n$, which are assumed to be uncorrelated zero-mean Weiner processes with standard deviations $\sigma_v$ and $\sigma_\omega$. The full stochastic motion model for the robot is described by

$$d\begin{bmatrix} x_r \\ y_r \\ z_r \\ \phi_r \end{bmatrix} = \begin{bmatrix} v_r\cos\phi_r \\ v_r\sin\phi_r \\ 0 \\ \omega_r \end{bmatrix} dt + \begin{bmatrix} \sigma_v\cos\phi_r & 0 \\ \sigma_v\sin\phi_r & 0 \\ 0 & 0 \\ 0 & \sigma_\omega \end{bmatrix} \begin{bmatrix} dv_n \\ d\omega_n \end{bmatrix} \tag{2}$$

The robot moves through an environment populated by $n$ point landmarks, which it is capable of observing through bearing measurements. We denote by $\mathcal{V}$ the set of all such landmarks and by $\mathcal{G}(t) \subset \mathcal{V}$ the set of landmarks observable at time $t$. Let the Cartesian coordinates of the $i$th landmark be denoted by $\mathbf{p}_i = [x_i \; y_i \; z_i]^{\mathrm{T}}$. Then the true measurements of the $i$th landmark can be expressed as

$$\begin{array}{rcl} \alpha_i &=& \arctan\dfrac{y_i - y_r}{x_i - x_r} - \phi_r \\ \beta_i &=& \arcsin\dfrac{z_i - z_r}{\sqrt{(x_i-x_r)^2 + (y_i-y_r)^2 + (z_i-z_r)^2}} \end{array} \tag{3}$$

or concisely using the measurement vector $\mathbf{y}_i(t) = [\alpha_i(t) \; \beta_i(t)]^\top$. Let $\mathbf{z} = [\mathbf{x}_r^\top \; \mathbf{p}_i^\top \; \ldots \; \mathbf{p}_n^\top]^\top$ denote a traditional SLAM state vector. The measurements $\mathbf{y}_i(t)$ are typically corrupted by a noise process $\mathbf{n}(t)$ such that

$$d\mathbf{y}(t) \triangleq \psi dt = h(\mathbf{z})dt + \mathbf{E}(t)\mathbf{n}(t) \tag{4}$$

in continuous-time. Here, we assume that $\mathbf{n}(t)$ is a zero-mean Weiner process and $\mathbf{E}(t)$ is a measurement noise weighting matrix that can be dependent on the true state. The measurements and robot dynamics are nonlinear in the chosen Cartesian coordinate system.

## III. Robocentric Mapping in Modified Spherical Coordinates

The contribution of this paper is a novel robocentric algorithm for mapping and localization which takes advantage of the spherical-like nature of the relative bearing measurements. There does not appear to be any similar (spherical-like) algorithms in the SLAM or robocentric mapping literature.

The spherical coordinates of landmark $i$ in the robot's reference frame is given by

$$\begin{array}{rcl} \alpha_i &=& \arctan\dfrac{y_i - y_r}{x_i - x_r} - \phi_r \\ \beta_i &=& \arcsin\dfrac{z_i - z_r}{\sqrt{(x_i-x_r)^2 + (y_i-y_r)^2 + (z_i-z_r)^2}} \\ d_i &=& \sqrt{(x_i-x_r)^2 + (y_i-x_r)^2 + (z_i-z_r)^2} \end{array} \tag{5}$$

which we write succinctly as $\mathbf{r}_i = [\alpha_i \; \beta_i \; d_i]^\top$. Using (5) together with the unicycle motion model (1) of the robot yields

$$\begin{array}{rcl} \dot{\alpha}_i &=& \dfrac{v_r \sin\alpha_i}{d_i\cos\beta_i} - \omega_r \\ \dot{\beta}_i &=& \dfrac{v_r}{d_i}\cos\alpha_i\sin\beta_i \\ \dot{d}_i &=& -v_r\cos\alpha_i\cos\beta_i \end{array} \tag{6}$$

Taking the previously defined noise processes $v_n$ and $\omega_n$ into account, we get the following stochastic motion model

$$d\begin{bmatrix} \alpha_i \\ \beta_i \\ d_i \end{bmatrix} = \begin{bmatrix} \frac{v_r\sin\alpha_i}{d_i\cos\beta_i} - \omega_r \\ \frac{v_r}{d_i}\cos\alpha_i\sin\beta_i \\ -v_r\cos\alpha_i\cos\beta_i \end{bmatrix} dt + \begin{bmatrix} \frac{\sigma_v\sin\alpha_i}{d_i\cos\beta_i} & -\sigma_\omega \\ \frac{\sigma_v}{d_i}\cos\alpha_i\sin\beta_i & 0 \\ -\sigma_v\cos\alpha_i\cos\beta_i & 0 \end{bmatrix} \begin{bmatrix} dv_n \\ d\omega_n \end{bmatrix} \tag{7}$$

We thus have a nonlinear system (7) and linear measurements (9). However, we go one step further and modify the dynamic system (7) slightly. In particular, we will not consider the range $d_i$ of each landmark $i$ but rather the inverse range $\rho_i = 1/d_i$, see [5], [13]. In this case, we have $\dot{\rho}_i = -\rho_i^2\dot{d}_i$ or taking account of the input noise we have

$$d\rho_i = v_r\rho_i^2\cos\alpha_i\cos\beta_i dt + \sigma_v\rho_i^2\cos\alpha_i\cos\beta_i dv_n \tag{8}$$

and thus the modification of the dynamic system (7) is obvious. The reason for using $\rho_i$ instead of $d_i$ is related to the initialization and is explained in the subsequent subsection. We redefine $\mathbf{r}_i = [\alpha_i \; \beta_i \; \rho_i]^\top$ and $\mathbf{z} = [\mathbf{r}_1 \; \ldots \; \mathbf{r}_n]^\top$. The measurements (3) are linear in $\mathbf{r}_i$ or more generally in $\mathbf{z} = [\mathbf{r}_1 \; \ldots \; \mathbf{r}_n]^\top$ and can then be given by the continuous-time measurement equation

$$d\mathbf{y}(t) \triangleq \psi dt = \mathbf{H}(\mathcal{G}(t))\mathbf{z}dt + \mathbf{E}(t)\mathbf{n}(t) \tag{9}$$

where $\mathbf{E}(t)$ is not required to be independent of $\mathbf{z}$. Here, $\mathbf{H}(\mathcal{G}(t))$ is a time-varying linear matrix which is dependent only on the set $\mathcal{G}(t)$ of currently sensed landmarks.

The function $f_i(\cdot)$ which captures the dynamics of the subspace $\mathbf{r}_i$ is

$$f_i(\widehat{\mathbf{z}}, v_r, w_r) = f_i(\widehat{\mathbf{r}}_i, v_r, w_r) = \begin{bmatrix} \frac{v_r\rho_i\sin\alpha_i}{\cos\beta_i} - \omega_r \\ v_r\rho_i\cos\alpha_i\sin\beta_i \\ v_r\rho_i^2\cos\alpha_i\cos\beta_i \end{bmatrix} \tag{10}$$

and where $f(\cdot)$ is thus a vertical concatenation of the $f_i(\cdot)$. For latter use we introduce the following Taylor expansion of $f(\cdot)$ about the estimate $\widehat{\mathbf{z}}$,

$$f(\mathbf{z}, v_r, w_r) - f(\widehat{\mathbf{z}}, v_r, w_r) = \mathbf{A}(t)(\mathbf{z} - \widehat{\mathbf{z}}) + \varrho(\mathbf{z}, \widehat{\mathbf{z}}, v_r, w_r) \tag{11}$$

where $\mathbf{A}(t)$ is the Jacobian of $f(\cdot)$ and $\varrho(\mathbf{z}, \widehat{\mathbf{z}}, v_r, w_r)$ accounts for the higher order terms. The Jacobian $\mathbf{A}_i(t)$ of $f_i(\cdot)$ is given by

$$\mathbf{A}_i = v_r\rho_i^2 \begin{bmatrix} \frac{\cos\alpha_i}{\rho_i\cos\beta_i} & \frac{\sin\alpha_i\sin\beta_i}{\rho_i\cos^2\beta_i} & \frac{\sin\alpha_i}{\rho_i^2\cos\beta_i} \\ -\frac{\sin\alpha_i\sin\beta_i}{\rho_i} & \frac{\cos\alpha_i\cos\beta_i}{\rho_i} & \frac{\cos\alpha_i\sin\beta_i}{\rho_i^2} \\ -\sin\alpha_i\cos\beta_i & -\cos\alpha_i\sin\beta_i & \frac{2\cos\alpha_i\cos\beta_i}{\rho_i} \end{bmatrix} \tag{12}$$

and is evaluated at an estimate $\widehat{\mathbf{r}}_i$. For any time-varying matrix $\mathbf{M}(t)$ we introduce the following notation

$$\|\mathbf{M}(t)\| = \sup\{\|\mathbf{M}(t)\| : m_{ij} \in \mathbb{R}\} \tag{13}$$

for all $t$ and for some norm $\|\cdot\|$. Moreover, we make the following standing assumption for simplicity.

*Assumption 1: The robot does not travel directly over or directly underneath a true landmark location or the estimated location of a landmark, i.e.* $\beta_i \neq \pm\pi/2$ *or* $\widehat{\beta}_i \neq \pm\pi/2$.

Assumption 1 is a technical requirement of the chosen coordinate system but not strong in practice. In fact, landmarks are often not chosen automatically to lie directly above or below the robot's trajectory and if indeed they were then we could subsequently alter the robot trajectory to avoid this. As a consequence of the assumption, the following bound holds

$$\|\mathbf{A}(t)\| = \overline{a} < \infty \tag{14}$$

for all $t$ given a particular robot trajectory. The error $\zeta = \mathbf{z} - \widehat{\mathbf{z}}$ is the so-called state estimation error. We will also need the following lemma concerning the growth of $\varrho(\mathbf{z}, \widehat{\mathbf{z}}, v_r, w_r)$.

*Lemma 1: The following inequality holds*

$$\begin{aligned} \|\varrho(\mathbf{z}, \widehat{\mathbf{z}}, v_r, w_r)\| &= \|f(\mathbf{z}, \cdot) - f(\widehat{\mathbf{z}}, \cdot) - \mathbf{A}(t)(\mathbf{z} - \widehat{\mathbf{z}})\| \\ &\leq 2\overline{a}\|\zeta\| \end{aligned} \tag{15}$$

*for with probability* 1 *when Assumption 1 holds.*

*Proof:* From the triangle inequality we obtain

$$\begin{aligned} \|f(\mathbf{z}, v_r, w_r) - f(\widehat{\mathbf{z}}, v_r, w_r) - \mathbf{A}(t)(\mathbf{z} - \widehat{\mathbf{z}})\| &\leq \\ \|f(\mathbf{z}, v_r, w_r) - f(\widehat{\mathbf{z}}, v_r, w_r)\| + \| - \mathbf{A}(t)\zeta\| &\leq \\ \|f(\mathbf{z}, v_r, w_r) - f(\widehat{\mathbf{z}}, v_r, w_r)\| + \overline{a}\zeta \end{aligned} \tag{16}$$

which follows using (14). Now if $f(\cdot)$ is Lipschitz then $\|f(\mathbf{z}, v_r, w_r) - f(\widehat{\mathbf{z}}, v_r, w_r)\| \leq c\|\mathbf{z} - \widehat{\mathbf{z}}\|$ for some $0 < c < \infty$. Actually, we know that if $\|\mathbf{A}(t)\|$ is bounded by $\overline{a}$ then $f(\cdot)$ is Lipschitz with Lipschitz coefficient $\overline{a}$. ■

Note also that $\varrho(\mathbf{z}, \widehat{\mathbf{z}}, v_r, w_r) = 0$ when $\zeta(t) = 0$.

### A. Initialization in Modified Spherical Coordinates

Since the location of the landmarks, or even their number, is not known beforehand, we will need to augment our state with the parameters for each new landmark when they are first observed. This presents a problem, since we only observe the two bearings, $\alpha$ and $\beta$, in any single measurement, and not the depth (or inverse depth).

This problem is particularly severe in Cartesian coordinates, since the conic region of uncertainty will not be well approximated by a Gaussian distribution. One way to get around this is to use so called *delayed initialization* [16], where a landmark is not added to the state until it has been observed sufficiently for the depth to be estimated. However, this adds to the complexity of the implementation, requiring the provisional landmarks to be handled as a separate case.

Civera et al. [13] proposed a method for undelayed initialization for global SLAM by parameterizing the landmarks using the robot pose in Cartesian coordinates together with the two observed bearings and the inverse depth. In our robocentric formulation, we will only use the latter three parameters. The bearings are trivially initialized using the measurement values and measurement statistics. The uncertainty in the inverse depth can be reasonably well approximated using a Gaussian function [13]. Also, by using the inverse depth $\rho_i$ instead of $d_i$, we can better account for a very large range of initial $d_i$ values (including $\infty$) in the uncertainty region given a reasonable value for $\rho_i(0)$ and $\sigma_\rho(0)$.

### B. Observability of the Proposed Estimation Problem

When discussing the observability of the proposed SLAM algorithm, we will use the property of *local weak observability*, as defined by Hermann and Krener in [17]. The same method was previously applied to the global SLAM problem in [8] to prove its fundamental unobservability.

A system $\Sigma$ is said to be *locally weakly observable* at a point $\mathbf{x}^0$ if we can instantaneously distinguish $\mathbf{x}^0$ from its neighbors. To test for this property, we use the *observability rank condition*, which is a sufficient condition for local weak observability. For simplicity, we will only consider a system with a single landmark $i$. We define our system $\Sigma$ as

$$\Sigma: \quad \begin{aligned} \dot{\mathbf{x}} = f(\mathbf{x}, v_r, \omega_r) &= \begin{pmatrix} f_\alpha \\ f_\beta \\ f_\rho \end{pmatrix} = \begin{pmatrix} \frac{v_r \rho_i \sin\alpha_i}{\cos\beta_i} - \omega_r \\ v_r \rho_i \cos\alpha_i \sin\beta_i \\ v_r \rho_i^2 \cos\alpha_i \cos\beta_i \end{pmatrix} \\ \mathbf{y} = h(\mathbf{x}) &= \begin{pmatrix} h_1 \\ h_2 \end{pmatrix} = \begin{pmatrix} \alpha_i \\ \beta_i \end{pmatrix} \end{aligned}$$

where $\mathbf{x} \in \mathcal{M}$ and $\mathcal{M}$ is a 3-dimensional $C^\infty$ connected manifold and $f$ and $h$ are $C^\infty$ functions.

Let $\mathrm{O}_\Sigma$ be the matrix whose rows consist of repeated Lie derivatives of one-forms $dh_i(\mathbf{x})$ with respect to the Lie algebra $\mathcal{F}$ of vector fields generated by $f(\mathbf{x}, v_r, \omega_r)$ on $\mathcal{M}$. These repeated Lie derivatives are defined recursively as

$$L_f^0 dh_i(\mathbf{x}) = \frac{\partial h_i(\mathbf{x})}{\partial \mathbf{x}} \tag{17}$$

and given an iterative index $q \in \mathbb{N}$ we have

$$\begin{aligned} L_f^q dh_i(\mathbf{x}) &= L_f^{q-1} dh_i(\mathbf{x}) \tfrac{\partial f(\mathbf{x}, v_r, \omega_r)}{\partial \mathbf{x}} \\ &\quad + \left[ \tfrac{\partial}{\partial \mathbf{x}} \left( L_f^{D-1} dh_i(\mathbf{x}) \right)^{\mathrm{T}} f(\mathbf{x}, v_r, \omega_r) \right]^{\mathrm{T}} \end{aligned} \tag{18}$$

For our system $\Sigma$, we have

$$L_f^0 dh_1 = [1\ 0\ 0], \qquad L_f^0 dh_2 = [0\ 1\ 0] \tag{19}$$

$$L_f^1 dh_1 = v_r \begin{bmatrix} \frac{\rho_i \cos\alpha_i}{\cos\beta_i} & \frac{\rho_i \sin\alpha_i \sin\beta_i}{\cos^2\beta_i} & \frac{\sin\alpha_i}{\cos\beta_i} \end{bmatrix} \tag{20}$$

$$L_f^1 dh_2 = v_r \rho_i [\frac{-\sin\alpha_i \sin\beta_i}{v_r} \ \cos\alpha_i \cos\beta_i \ \cos\alpha_i \sin\beta_i] \tag{21}$$

$$L_f^2 dh_1(\mathbf{x}) = \begin{bmatrix} \frac{4v_r^2\rho_i^2\cos^2\alpha_i}{\cos^2\beta_i} - \frac{2v_r^2\rho_i^2}{\cos^2\beta_i} + \frac{v_r\omega_r\rho_i\sin\alpha_i}{\cos\beta_i} \\ \frac{4v_r^2\rho_i^2\sin\alpha_i\cos\alpha_i\sin\beta_i}{\cos^2\beta_i} - \frac{v_r\omega_r\rho_i\cos\alpha_i\sin\beta_i}{\cos\beta_i} \\ \frac{4v_r^2\rho_i\sin\alpha_i\cos\alpha_i}{\cos^2\beta_i} - \frac{v_r\omega_r\cos\alpha_i}{\cos\beta_i} \end{bmatrix}^{\top} \tag{22}$$

$$L_f^2 dh_2(\mathbf{x}) = \begin{bmatrix} \frac{2v_r^2\rho_i^2 \sin\alpha_i \cos\alpha_i \sin\beta_i}{\cos\beta_i} - 4v_r^2\rho_i^2 \sin\alpha_i \cos\alpha_i \sin\beta_i \cos\beta_i + v_r\omega_i\rho_i \cos\alpha_i \sin\beta_i \\ 4v_r^2\rho_i^2 \cos^2\alpha_i \cos^2\beta_i - 2v_r^2\rho_i^2 \cos^2\alpha_i - \frac{v_r^2\rho_i^2 \sin^2\alpha_i}{\cos^2\beta_i} + v_r\omega_r \cos\alpha_i \sin\beta_i \\ 4v_r^2\rho_i \cos^2\alpha_i \sin\beta_i \cos\beta_i - \frac{2v_r^2\rho_i \sin^2\alpha_i \sin\beta_i}{\cos\beta_i} + v_r\omega_r \sin\alpha_i \sin\beta_i \end{bmatrix}^\top \tag{23}$$

If $\mathrm{rank}(\mathrm{O}_\Sigma) = 3$, for some point $\mathbf{x}^0$, the system fulfills the observability rank condition at this point, and is thus locally weakly observable at this point [17].

Note by inspection, if $v_r = 0$, $\mathrm{O}_\Sigma$ will never be full rank since the third column will be zero. Intuitively, the robot is stationary and can never observe any change in the landmark bearings and thus cannot observe depth. If $v_r \neq 0$ and $\omega_r = 0$ then (20) and (21) ensure that the observability rank condition will hold as long as $\alpha_i$ and $\beta_i$ are not both equal to zero. This means that $\Sigma$ will be locally weakly observable for all $\mathbf{x}$ with $\alpha_i \neq 0$ and $\beta_i \neq 0$. This also agrees with intuition, since if the robot was moving directly towards the landmark, it would not observe any change in bearing.

Finally, if both $v_r \neq 0$ and $\omega_r \neq 0$ then (22) ensures that the observability rank condition will hold for every $\mathbf{x} \in \mathcal{M}$ and thus $\Sigma$ will be locally weakly observable. Intuitively, this corresponds to the robot traveling on an arc, so that no landmark will remain on the indistinguishable line.

### C. The Estimator

The behavior of an estimate $\widehat{\mathbf{z}}$ of $\mathbf{z}$ depends on the particular estimator. Thus we consider an estimator of the form

$$d\widehat{\mathbf{z}} = f(\widehat{\mathbf{z}}, v_r, w_r)dt + \mathbf{K}(t)\left(d\mathbf{y}(t) - \mathbf{H}(\mathcal{G}(t))\widehat{\mathbf{z}}dt\right) \tag{24}$$

The gain $\mathbf{K}(t)$ is given by

$$\mathbf{K}(t) = \mathbf{P}(t)\mathbf{H}^\top(\mathcal{G}(t))\mathbf{R}^{-1}(t) \tag{25}$$

and $\mathbf{P}(t)$ is the solution to the following Riccati differential equation

$$\begin{aligned} d\mathbf{P}(t) &= \left[\mathbf{A}(t)\mathbf{P}(t) + \mathbf{P}(t)\mathbf{A}^\top(t) + \mathbf{Q}(t)\right]dt - \\ &\quad \mathbf{P}(t)\mathbf{H}^\top(\mathcal{G}(t))\mathbf{R}^{-1}(t)\mathbf{H}(\mathcal{G}(t))\mathbf{P}(t) \end{aligned} \tag{26}$$

where $\mathbf{Q}$ and $\mathbf{R}$ are positive-definite tuning matrices. For completeness we state the following common assumption.

*Assumption 2:* The following $\mathbf{Q}(t) \geq \underline{q}\mathbf{I}$, $\mathbf{R}(t) \geq \underline{r}\mathbf{I}$ and $\mathbf{P}(t_0) \geq p_0\mathbf{I}$ are given for some $\underline{q}, \underline{r}, p_0 > 0$ such that $\|\mathbf{Q}(t)\| \geq \underline{q}$ and $\|\mathbf{R}(t)\| \geq \underline{r}$. Moreover, $\mathbf{Q}(t)$ and $\mathbf{R}(t)$ are chosen to be bounded by $\|\mathbf{Q}(t)\| \leq \overline{q} < \infty$ and $\|\mathbf{R}(t)\| \leq \overline{r} < \infty$ for all $t$. Also, we have $\mathbf{E}(t) \leq \overline{e} < \infty$ with $\mathbf{E}(t) \geq \underline{e}\mathbf{I}$.

The analysis in this paper will consider the propagation of the estimation error $\zeta(t) = \mathbf{z}(t) - \widehat{\mathbf{z}}(t)$ for all $t > t_0$ given an initial estimation error $\zeta(t_0)$ which we will assume belongs to the set

$$\|\zeta(t_0)\| \leq b \quad \text{in the state space} \tag{27}$$

for some constant $b < \infty$. We assume initially that $\widetilde{\mathcal{G}}(t) = \mathcal{G}$ for all $t > t_0$. It is common to assume a full landmark measurement vector for analysis [3], [11].

Note that in general, the continuous time estimator in this section does not involve a prediction stage. However, by letting $\mathbf{R}^{-1}(t) = 0$ over the time interval $t \in [k_0, k_1]$ we can easily allow for the absence of measurements over that interval.

### D. On the Convergence of the Feature Estimator

We consider (for simplicity) the case where $\tilde{\mathcal{G}}(t) = \mathcal{G}(t)$ for all $t$. We assume an EKF-like algorithm of the form (24) with Assumptions 1-2 holding. The error $\zeta = \mathbf{z} - \widehat{\mathbf{z}}$ obeys

$$\begin{aligned} d\zeta &= \left[(\mathbf{A}(t) - \mathbf{K}(t))\,\zeta + \varrho(\mathbf{z}, \widehat{\mathbf{z}}, v_r, w_r)\right]dt + \\ &\quad \mathbf{G}(t)\begin{bmatrix} dv_n \\ dw_n \end{bmatrix} - \mathbf{K}(t)\mathbf{E}(t)d\mathbf{n}(t) \\ d\zeta &= \left[(\mathbf{A}(t) - \mathbf{K}(t))\,\zeta + \varrho(\mathbf{z}, \widehat{\mathbf{z}}, v_r, w_r)\right]dt + \\ &\quad \left[\mathbf{G}(t) \;\; -\mathbf{K}(t)\mathbf{E}(t)\right]\begin{bmatrix} \begin{bmatrix} dv_n \\ dw_n \end{bmatrix} \\ d\mathbf{n}(t) \end{bmatrix} \end{aligned} \tag{28}$$

where $\mathbf{G}_i(t)$ is given by

$$\|\mathbf{G}_i(t)\| = \left\| \begin{bmatrix} \frac{\sigma_v\rho_i \sin\alpha_i}{\cos\beta_i} & -\sigma_\omega \\ \rho_i\sigma_v \cos\alpha_i \sin\beta_i & 0 \\ -\sigma_v\rho_i^2 \cos\alpha_i \cos\beta_i & 0 \end{bmatrix} \right\| = \overline{g} < \infty \tag{29}$$

and $\mathbf{G}(t) = \begin{bmatrix}\mathbf{G}_1^\top & \dots & \mathbf{G}_n^\top\end{bmatrix}^\top$. Recall Lemma 1 bounds the growth of the nonlinear perturbation term $\varrho(\mathbf{z}, \widehat{\mathbf{z}}, v_r, w_r)$. We need the following assumption.

*Assumption 3: The state covariance $\mathbf{P}(t)$ is bounded by*

$$0 < \underline{p} \leq \mathbf{P}(t) \leq \overline{p} < \infty \tag{30}$$

*for all $t > t_0$*

Note that Assumption 3 is quite reasonable. In fact, the lower bound follows from a general controllability argument. We also conjecture based on the analysis in [18] that it is possible to formally prove that Assumption 3 holds for all $t$, given only that the control inputs ensure the state is observable, e.g. $v_r \neq 0$ for all $t$, and $\|\mathbf{A}(t)\|$ is bounded.

*Theorem 1:* Consider (28) with an initial condition (27) and $\mathcal{G}(t) = \mathcal{V}(t)$ for all $t$. Suppose that Assumptions 1-3 hold. If

$$\|\mathbf{P}^{-1}(t)\mathbf{Q}(t)\mathbf{P}^{-1}(t) + \mathbf{R}^{-1}(t)\|\underline{p} > 4\overline{ap}/\underline{p} \tag{31}$$

then the estimation error is bounded above with

$$\mathcal{E}\left\{\|\zeta(t)\|^2\right\} \leq \max\left\{\frac{n(\underline{r}^2\overline{g}^2 + \overline{p}\underline{p}\overline{e}^2)}{2\gamma\underline{r}^2\underline{p}},\ \frac{\overline{p}}{\underline{p}}\|\zeta(t_0)\|^2\right\} \tag{32}$$

where

$$\gamma = \|\mathbf{P}^{-1}(t)\mathbf{Q}(t)\mathbf{P}^{-1}(t) + \mathbf{R}^{-1}(t)\|\underline{p} - \frac{4\overline{ap}}{\underline{p}} \tag{33}$$

and $\mathcal{E}\left\{\|\zeta(t)\|^2\right\}$ as $t \to \infty$ is bounded by $\frac{n(\underline{r}^2\overline{g}^2 + \overline{p}\underline{p}\overline{e}^2)}{2\gamma\underline{r}^2\underline{p}}$.

*Proof:* Let $\mathcal{B}(t,\zeta(t)) = \zeta^\top(t)\mathbf{P}^{-1}(t)\zeta(t) > 0$ and note that

$$\begin{aligned} d\mathcal{B} &= \left[\frac{\partial \mathcal{B}}{\partial t} + \frac{\partial \mathcal{B}}{\partial \zeta}\left(\mathbf{A}(t) - \mathbf{K}(t)\right)\zeta + \frac{\partial \mathcal{B}}{\partial \zeta}\varrho(\mathbf{z},\widehat{\mathbf{z}},\cdot)\right]dt + \\ &\quad \frac{1}{2}\mathrm{tr}\left(\mathrm{hess}(\mathcal{B})\boldsymbol{\Xi}(t)\boldsymbol{\Xi}^\top(t)\right)dt - \\ &\quad \frac{\partial \mathcal{B}}{\partial \zeta}\boldsymbol{\Xi}(t)\left[\begin{array}{c}\left[\begin{array}{c} dv_n \\ dw_n \end{array}\right] \\ d\mathbf{n}(t)\end{array}\right] \end{aligned} \tag{34}$$

where $\boldsymbol{\Xi}(t) = [\mathbf{G}(t) \ -\mathbf{K}(t)\mathbf{E}(t)]$ and where we have employed Itos differential formula. Evaluating the terms and rearranging leads to

$$\begin{aligned} d\mathcal{B} &= \left[\zeta^\top\left[\mathbf{P}^{-1}(t)\mathbf{Q}(t)\mathbf{P}^{-1}(t) + \mathbf{R}^{-1}(t)\right]\zeta\right]dt + \\ &\quad 2\zeta^\top\mathbf{P}^{-1}(t)\varrho(\mathbf{z},\widehat{\mathbf{z}},v_r,w_r)dt + \\ &\quad \frac{1}{2}\mathrm{tr}\left(\mathbf{P}^{-1}(t)\mathbf{G}(t)\mathbf{G}^\top(t)\right)dt + \\ &\quad \frac{1}{2}\mathrm{tr}\left(\mathbf{P}^{-1}(t)\mathbf{K}(t)\mathbf{K}^\top(t)\right)dt - \\ &\quad 2\zeta^\top\mathbf{P}^{-1}(t)\boldsymbol{\Xi}(t)\left[\begin{array}{c}\left[\begin{array}{c} dv_n \\ dw_n \end{array}\right] \\ d\mathbf{n}(t)\end{array}\right] \\ &\leq \left[-\alpha\|\zeta\|^2 + \frac{4\overline{a}}{\underline{p}}\|\zeta\|^2 + \frac{n(\underline{r}^2\overline{g}^2 + \overline{p}\underline{p}\overline{e}^2)}{2\gamma\underline{r}^2\underline{p}}\right]dt - \\ &\quad 2\zeta^\top\mathbf{P}^{-1}(t)\boldsymbol{\Xi}(t)\left[\begin{array}{c}\left[\begin{array}{c} dv_n \\ dw_n \end{array}\right] \\ d\mathbf{n}(t)\end{array}\right] \end{aligned} \tag{35}$$

where we have explicitly employed Lemma 1 and where

$$\alpha = \|\mathbf{P}^{-1}(t)\mathbf{Q}(t)\mathbf{P}^{-1}(t) + \mathbf{R}^{-1}(t)\| \tag{36}$$

Now noting that $\overline{p}^{-1}\|\zeta\|^2 \leq \mathcal{B}(t,\zeta(t)) \leq \underline{p}^{-1}\|\zeta\|^2$ and using the Bellman-Gromwall lemma [19] we come to

$$\begin{aligned} \mathcal{B}(t,\zeta(t)) &\leq \mathcal{B}(t_0,\zeta(t_0))\exp\left(-\gamma(t-t_0)\right) - \\ &\quad 2\int_{t_0}^{t}\zeta^\top(\tau)\mathbf{P}^{-1}(\tau)\boldsymbol{\Xi}(\tau)\left[\begin{array}{c}\left[\begin{array}{c} dv_n(\tau) \\ dw_n(\tau) \end{array}\right] \\ d\mathbf{n}(\tau)\end{array}\right] + \\ &\quad \frac{n(\underline{r}^2\overline{g}^2 + \overline{p}\underline{p}\overline{e}^2)}{2\gamma\underline{r}^2\underline{p}} - \\ &\quad \frac{n(\underline{r}^2\overline{g}^2 + \overline{p}\underline{p}\overline{e}^2)}{2\gamma\underline{r}^2\underline{p}}\exp\left(-\gamma(t-t_0)\right) \end{aligned} \tag{37}$$

where $\gamma = \left(\alpha\underline{p} - 4\frac{\overline{a}\overline{p}}{\underline{p}}\right)$ with $\gamma > 0$ if and only if $\alpha\underline{p} > \frac{4\overline{a}\overline{p}}{\underline{p}}$. Taking the expectation of (49) and rearranging gives

$$\mathcal{E}\left\{\|\zeta(t)\|^2\right\} \leq \max\left\{\frac{n(\underline{r}^2\overline{g}^2 + \overline{p}\underline{p}\overline{e}^2)}{2\gamma\underline{r}^2\underline{p}},\ \frac{\overline{p}}{\underline{p}}\|\zeta(t_0)\|^2\right\} \tag{38}$$

for all $t$ if $\gamma > 0$. The error $\mathcal{E}\left\{\|\zeta(t)\|^2\right\}$ as $t \to \infty$ is bounded by $\frac{n(\underline{r}^2\overline{g}^2+\overline{p}\underline{p}\overline{e}^2)}{2\gamma\underline{r}^2\underline{p}}$. This completes the proof. ■

Importantly, we have provided conditions under which an EKF-like algorithm will yield an exponentially bounded and converging mean-square estimation error.

## IV. Numerical Analysis

The proposed algorithm is now illustrated via simulation. We compare the performance of the spherical robot-centric SLAM algorithm against a global SLAM algorithm similar to the one used in [13] in a simulated environment.

The state of the global SLAM algorithm takes the form $\mathbf{g} = [x_r \ y_r \ z_r \ \phi_r \ \mathbf{b}_1^\top \ \dots \ \mathbf{b}_n^\top]^\top$ with $\mathbf{b}_i = [x_{ir}^* \ y_{ir}^* \ z_{ir}^* \ \alpha_i^* \ \beta_i^* \ \rho_i^*]^\top$ for a single landmark $i$. The $x_{ir}^*$, $y_{ir}^*$ and $z_{ir}^*$ state components are the position of the robot in global coordinates when the landmark $i$ is first initialized (or observed). Here $\alpha_i^*$, $\beta_i^*$ and $\rho_i^*$ are the $\alpha_i$, $\beta_i$ and $\rho_i$ values relative to the robot's position when landmark $i$ is first initialized, i.e. $[x_{ir}^* \ y_{ir}^* \ z_{ir}^*]^\top$ and the position of landmark $i$. The reason for this parametrization [13] is that it enables us to do single step initialization of the landmarks, which would be very difficult using a purely Cartesian representation.

A typical map and robot trajectory is shown in Fig 1. In each simulation run, the landmarks are distributed randomly in a square pattern along the $x$ and $y$ axes, and uniformly on the $z$-axis. The robot moves along a randomly generated path in the center of the environment.

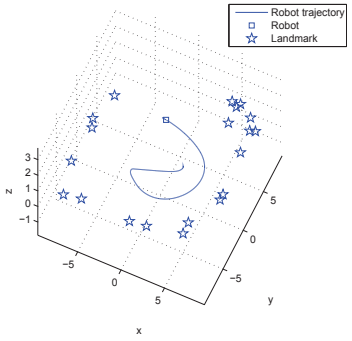



Fig. 1. A typical map layout and robot trajectory.

### A. Example Scenario

In this case, the robot can sense landmark $i$ if and only if $\alpha_i(t) \in (-\pi/4, \pi/4)$ and $\beta_i(t) \in (-\pi/4, \pi/4)$. The process noise is $\sigma_v = 0.42$, $\sigma_\omega = 1.06$ and the measurement noise has standard deviation $0.0873$ radians for both bearings. Each simulation runs for $1500$ time steps.

Since the robocentric algorithm does not estimate the robot's global position, we cannot compare the trajectories produced by the two algorithms. Thus, for the global algorithm, we generate a relative map by computing the estimated spherical landmark position relative to the estimated robot pose. The RMS errors are shown in Fig 2 over 1000 simulations. For both algorithms, the errors converge to a steady value and both perform well in terms of the relative map.

Note that Fig 2 does not illustrate the error in the robot pose or its effect on the global map. In some simulations, both the global map and the pose estimate were offset by a significant amount. Fig 3 shows a birds-eye view of such a case and highlights the flaws of world-centric mapping.

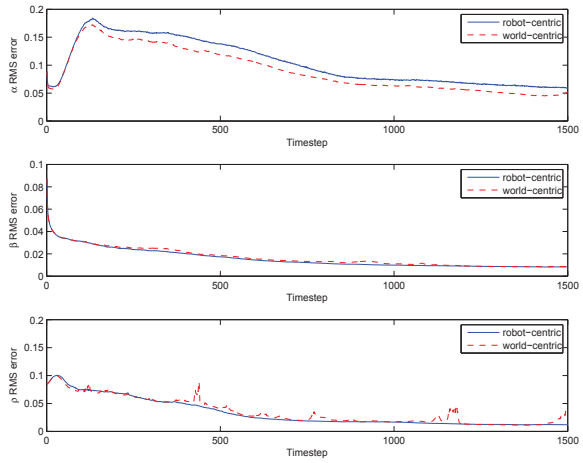



Fig. 2. The RMS error for the relative $\alpha$ bearings, $\beta$ bearings and inverse distance $\rho$ respectively.

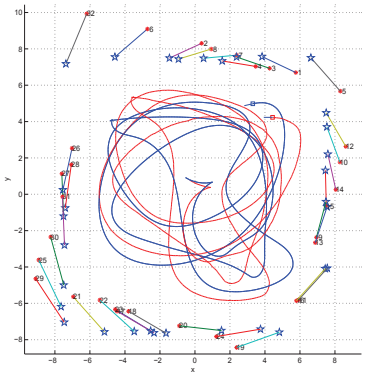


Fig. 3. This figure shows the global SLAM map and estimated robot trajectory is rotated and displaced quite significantly. The error in the global coordinates occurs despite, as shown in Fig 2, the fact that an accurate relative map can be derived from the global state.

This distortion of the global map is caused by an unobservable state in global SLAM, as was shown in [8].

## V. A Note on Complexity

When analyzing the computational complexity of a SLAM algorithm, one critical factor is the size of the state. One of the costliest operations in the EKF is the multiplication of large matrices, which is typically $\mathcal{O}(m^3)$ for $m$-by-$m$ matrices.

This gives the spherical robot-centric algorithm a significant speed boost over the global SLAM algorithm, since the size of the state is effectively cut in half. This speed boost was also notably observed in every simulation.

## VI. Concluding Remarks

In this paper we proposed a computationally efficient robocentric mapping algorithm that can be implemented using a linear measurement equation. We further highlighted the problems caused by the unobservable state components in traditional, world-centric SLAM, i.e. Fig 3. We have illustrated (and suspect it is well known) that a relative map computed using the global SLAM state vector can perform comparably with a dedicated robocentric algorithm even if the actual global map is quite inaccurate. Given that only the relative state components are observable and the increased computational cost in maintaining a global map, we question the utility in doing so, and thus further motivate the work in this paper.

## References


[1] R. Smith, M. Self, and P. Cheeseman, "Estimating uncertain spatial relationships in robotics," *Autonomous Robot Vehicles*, pp. 167–193, Springer-Verlag 1990.

[2] S. Thrun, D. Fox, and W. Burgard, "A probabilistic approach to concurrent mapping and localization for mobile robots," *Mach. Learning Autonom. Robots*, vol. 31, pp. 29–53, 1998.

[3] M. Dissanayake, P. Newman, S. Clark, H. Durrant-Whyte, and M. Csorba, "A solution to the simultaneous localization and map building (SLAM) problem," *IEEE Transactions on Robotics and Automation*, vol. 17, no. 3, pp. 229–241, June 2001.

[4] M. Walter, R. Eustice, and J. Leonard, "Exactly sparse extended information filters for feature-based SLAM," *International Journal of Robotics Research*, vol. 26, no. 4, pp. 335–359, April 2007.

[5] V. Aidala and S. Hammel, "Utilization of modified polar coordinates for bearings-only tracking," *IEEE Transactions on Automatic Control*, vol. 28, no. 3, p. 283294, March 1983.

[6] T. Song and J. Speyer, "A stochastic analysis of a modified gain extended Kalman filter with applications to estimation with bearings only measurements," *IEEE Transactions on Automatic Control*, vol. 30, no. 10, p. 940949, October 1985.

[7] G. Huang, A. Mourikis, and S. Roumeliotis, "Analysis and improvement of the consistency of extended Kalman filter based SLAM," in *Proceedings of the 2008 International Conference on Robotics and Automation (ICRA)*, May 2008, pp. 473–479.

[8] K. W. Lee, W. Wijesoma, and I. Javier, "On the observability and observability analysis of SLAM," in *2006 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Oct. 2006, pp. 3569–3574.

[9] S. Julier and J. Uhlmann, "A counter example to the theory of simultaneous localization and map building," in *Proceedings of the 2001 International Conference on Robotics and Automation (ICRA)*, May 2001, pp. 4238–4243.

[10] T. Bailey, J. Nieto, J. Guivant, M. Stevens, and E. Nebot, "Consistency of the EKF-SLAM algorithm," in *Proceedings of the 2006 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, October 2006, p. 35623568.

[11] S. Huang and G. Dissanayake, "Convergence and consistency analysis for extended kalman filter based SLAM," *IEEE Transactions on Robotics*, vol. 23, no. 5, pp. 1036–1049, October 2007.

[12] J. Castellanos, R. Martinez-Cantin, J. Tardos, and J. Neira, "Robocentric map joining: Improving the consistency of EKF-SLAM," *Robotics and Autonomous Systems*, vol. 55, no. 1, pp. 21–29, January 2007.

[13] J. Civera, A. Davison, and J. Montiel, "Inverse depth parametrization for monocular SLAM," *IEEE Transactions on Robotics*, vol. 24, no. 5, pp. 932–945, Oct. 2008.

[14] B. Grocholsky, J. Keller, V. Kumar, and G. Pappas, "Cooperative air and ground surveillance," *IEEE Robotics & Automation Magazine*, vol. 13, no. 3, pp. 16–25, September 2006.

[15] A. Bishop, B. Fidan, B. Anderson, K. Dogancay, and P. Pathirana, "Optimality analysis of sensor-target geometries in passive localization: Part 1 - Bearing-only localization," in *Proceedings of the 3rd International Conference on Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP)*, December 2007.

[16] T. Bailey, "Constrained initialisation for bearing-only SLAM," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA'03)*, September 2003, pp. 1966–1971.

[17] R. Hermann and A. Krener, "Nonlinear controllability and observability," *IEEE Transactions on Automatic Control*, vol. Ac-22, pp. 728–740, October 1977.

[18] A. Bishop and P. Jensfelt, "A stochastically stable solution to the problem of robocentric mapping," in *Proc. of ICRA'09*, Kobe, Japan, 2009.

[19] S. Sastry, *Nonlinear Systems: Analysis, Stability and Control*. New York, N.Y.: Springer-Verlag, 1999.

# Global Robot Localization with Random Finite Set Statistics*

**Adrian N. Bishop**
National ICT Australia (NICTA)
Australian National University (ANU)
adrian.bishop@anu.edu.au

**Patric Jensfelt**
Center for Autonomous Systems
Royal Institute of Technology (KTH)
patric@kth.se

**Abstract –** *We re-examine the problem of global localization of a robot using a rigorous Bayesian framework based on the idea of random finite sets. Random sets allow us to naturally develop a complete model of the underlying problem accounting for the statistics of missed detections and of spurious/erroneously detected (potentially unmodeled) features along with the statistical models of robot hypothesis disappearance and appearance. In addition, no explicit data association is required which alleviates one of the more difficult sub-problems. Following the derivation of the Bayesian solution, we outline its first-order statistical moment approximation, the so called probability hypothesis density filter. We present a statistical estimation algorithm for the number of potential robot hypotheses consistent with the accumulated evidence and we show how such an estimate can be used to aid in re-localization of kidnapped robots. We discuss the advantages of the random set approach and examine a number of illustrative simulations.*

**Keywords:** Robot localization; multiple-hypothesis localization; PHD filtering; random-set-based localization.

## 1 Introduction

The general approach to global localization (when not using GPS or artificial beacons such as bar codes and transponders) is to compare information (or features) extracted from sensor readings with an a priori map associated with the global reference frame. Each comparison carries some evidence about where the robot may be, and the challenge is then, as efficiently as possible, to find the correct pose, or a number of poses, that are in some statistical sense the most consistent with the accumulated evidence.

The approach proposed in this work most closely resembles the multiple hypothesis localization algorithms such as [1–3]. For brevity, we must point to the literature [4] for information on the other common approaches; most notably the Monte-Carlo (particle-filter type) algorithms [5, 6].

In the multiple-hypothesis technique [1–3], a set of Gaussians is used to represent individual pose hypotheses. The advantage of this is that a standard Kalman based pose tracking module can be used to independently update each hypothesis in a simple scheme commonly known as multiple hypothesis tracking (MHT). A further advantage of using a set of Gaussians is that it enables us to explicitly reason about each hypothesis whereas a particle filter, or a position probability grid, in principle requires you to process a much larger number of particles/cells (and their weights/probabilities). This is computationally challenging and is often circumvented by performing thresholding on the probabilities or clustering to form fewer hypotheses. Further advantages of the MHT-based approaches is found in [2].

A disadvantage of the multi-hypothesis techniques is that they inherently don't solve some of the most difficult problems related to localization (albeit they do with external algorithm components). In particular, the data-association problem and the problem of estimating (in an integrated/optimal fashion) a meaningful statistic regarding the number of poses consistent with the accumulated evidence and system models are not inherent. The second problem is often not studied explicitly in most localization algorithms[1] but plays an important role in the localization performance when false positive feature detections occur and/or more general measurement/system models are considered (such as in this paper). In such cases, the former data-association problem is further complicated and so-called clutter-rejection must be incorporated.

Furthermore, the standard vector-based stochastic differential (or difference) equation framework is arguably not a natural formulation for the multi-hypothesis tracking problem. Instead, in this paper we will explicitly exploit the framework of random finite sets (RFS - a term made precise later) and stochastic point processes [7]. A theoretically op-

*A.N. Bishop was with the Royal Institute of Technology (KTH) in Stockholm, Sweden at the time of the original submission. The authors were supported by the Swedish Foundation for Strategic Research (SSF) through the Centre for Autonomous Systems (CAS) and by the EU project 'CogX'. A.N. Bishop was also supported by NICTA. NICTA is funded by the Australian Government as represented by the Department of Broadband, Communications and the Digital Economy and the Australian Research Council through the ICT Centre of Excellence program.

[1] Some algorithms can potentially provide an ad-hoc estimate on the number of consistent robot poses, e.g. by counting particle clusters or hypotheses above some weight. However, the framework discussed in this paper provides such an estimate inherently and, in particular, it provides an estimate of the mathematically *expected* number of hypotheses consistent with the data.

timal, Bayesian filtering, framework can then be formulated using the concept of finite set statistics (FISST) [8].

For computational reasons, Mahler [9] proposed a first-oder moment approximation to the full Bayesian solution and termed the first-moment, the probability hypothesis density (PHD). A generic sequential Monte Carlo implementation [10–12] has been proposed and accompanied by convergence results [12–14]. Alternatively, an analytic solution to the PHD recursion was presented in [15] for problems involving linear Gaussian target dynamics, a Gaussian birth model and linear Gaussian (partial) observations. It is shown in [15] that when the initial prior intensity is a Gaussian mixture, the posterior intensity at any subsequent time step is also a Gaussian mixture. Furthermore, the Gaussian-mixture PHD recursions can approximate the true posterior intensity to any desired degree of accuracy [16].

See [7, 8, 11] for a comprehensive background on random finite set-based estimation and the PHD filter.

### 1.1 Contribution

The PHD filter has primarily been examined in the context of target tracking (with various sensors etc) [7]. However, a recent paper [17] has examined the performance of the PHD filter in solving the simultaneous localization and mapping (SLAM) problem. The PHD filter-based SLAM implementation [17] was shown to outperform the base implementation of FastSLAM in a number of adverse environments. In this paper, we re-examine the problem of global robot localization using a rigorous Bayesian framework based on the concept of random finite sets and the PHD filter.

Random sets allow us to naturally develop a complete model of the underlying problem accounting for the statistics of missed detections and the statistics of spurious/erroneously detected (potentially unmodeled) features. In addition we incorporate the statistical models of robot hypothesis disappearance (death) and appearance (typically after kidnapping or error). No explicit data association is required which alleviates one of the more difficult subproblems. Following the derivation of a complete and integrated Bayesian solution, we outline its first-order statistical moment approximation, i.e. the probability hypothesis density filter. We present a statistical estimation algorithm for the expected number of potential robot hypotheses consistent with the accumulated evidence and we show how such an estimate can be used to aid in re-localization of kidnapped robots. We then discuss the advantages of the random set approach and examine a number of illustrative examples.

Our technique's ability to handle missed detections, false alarms (false feature detections) and to associate an entire set of measurement hypotheses to a set of robot pose hypotheses within an integrated framework is significant. The stochastic vector realizations of robot localization (including the traditional multiple hypothesis techniques [1–4]) have to deal with such problems explicitly and outside any Bayes optimal (or approximated) filter. To the best of the authors' knowledge, this paper presents the first complete Bayesian localization solution involving the concept of random finite sets and details the first implementation of the PHD filter for global robot localization.

## 2 Conceptual Model

The idea behind the set-based, multiple hypothesis generation and tracking technique presented in this paper is illustrated in Figure 1.

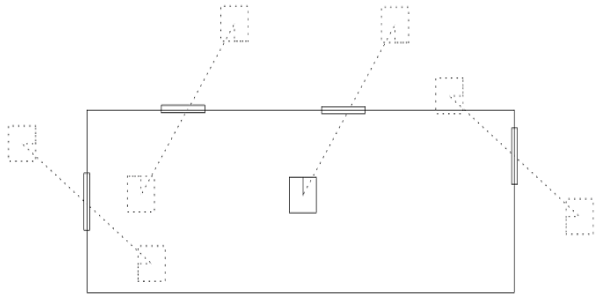


Figure 1: Multiple robot hypotheses are generated by a single measurement of some features in the environment given a priori knowledge that such features appear in a number of places in the global map.

In this illustration we see a situation where the true position of the robot is given by the solid square in the middle of the room in the figure. A door is detected in front and slightly to the right of the robot. Matching this feature to the map, consisting of four doors in one room, results in eight potential robot poses. These eight poses give rise to eight hypotheses regarding the pose of the robot. In the formulation outlined in this work, each hypothesis generated by a single feature detection is input to the estimation algorithm as an additional measurement. The grouping of such *measurement hypotheses* are modeled as random sets. In addition, the state of the robot's knowledge is modeled as a random set of *robot pose hypotheses*. The idea is that by making more observations of features in the environment and matching these to the existing robot pose hypotheses, those pose hypotheses which are not supported by the measurement set, i.e. the measurement hypotheses, can be eliminated (in a manner to be made precise) from the robot pose state set.

### 2.1 The Robot Set State Model

The true pose of a single robot is represented by the random variable $\mathbf{R}_t$ measured on the space $\mathcal{E} \subseteq \mathbb{R}^{n_r}$ with realization $\mathbf{r}_t$. The number of robot pose hypotheses is time-varying and given by $N_t$ at time $t$. We denote the individual pose hypotheses by $\mathbf{X}_t^i$. The set of state pose hypotheses at time $t$ is denoted by $\mathfrak{X}_t$.

The true state of a single robot is assumed to obey

$$\mathbf{R}_t = \psi_t\left(\mathbf{r}_{t-1}\right) + \mathbf{W}_t \tag{1}$$

where the input $\{\mathbf{W}_t\}$ is a sequence of independent Gaussian random variables that account for control input errors

and unmodeled dynamics etc. More general, non-Gaussian, error inputs can be accommodated in the general framework outlined in this paper.

Note that the transition density for the individual robot hypotheses $\mathbf{X}_t^i \in \mathfrak{X}_t$ is now given by

$$f_{t|t-1}^i(\mathbf{x}_t^i|\mathbf{x}_{t-1}^i) = \mathcal{N}(\mathbf{x}_t^i;\ \psi_t\left(\mathbf{x}_{t-1}^i\right), \mathbf{\Sigma}_t) \tag{2}$$

where $\mathcal{N}(\cdot;\ \mathbf{m}, \mathbf{P})$ denotes a Gaussian density function with mean $\mathbf{m}$ and covariance $\mathbf{P}$ and $\mathbf{\Sigma}_t$ is the covariance of $\mathbf{W}_t$.

The state set transition model in this paper incorporates statistical models of hypothesis appearance (birth) and hypothesis disappearance (death). New hypotheses might appear when the robot is kidnapped or in the case of localization error.

The probability that any hypothesis $i$ continues to exist at time $t$ given that it exists at $t-1$ is given by the survival probability $p_S^i$. Now it follows that

$$\mathfrak{X}_t \quad = \quad \left[\bigcup_{\mathbf{X}_{t-1}^i \in \mathfrak{X}_{t-1}} \mathfrak{S}_{t|t-1}(\mathbf{X}_{t-1}^i)\right] \cup b_t \mathfrak{B}_t \tag{3}$$

where $b_t \in \{0,1\}$ and $b_t\mathfrak{B}_t$ is defined to be $\mathfrak{B}_t$ when $b_t = 1$ or $\emptyset$ when $b_t = 0$. Also,

$$\mathfrak{S}_{t|t-1}(\mathbf{X}_{t-1}^i) = \begin{cases} \mathbf{X}_t^i \cap \mathcal{E} & \text{with probability } p_S^i \\ \emptyset & \text{with probability } 1 - p_S^i \end{cases} \tag{4}$$

where the evolution of $\mathbf{X}_{t-1}^i$ follows (2). If we neglect $\mathfrak{B}_t$, then it is clear that we are modeling the motion and death of a number of robot hypotheses in (3). That is, if $b_t = 0$ then our set transition model accounts only for random hypothesis disappearance.

The input $b_t \in \{0,1\}$ acts as a kidnapped robot switch which is set to 1 based on the outcome of kidnapped robot test outlined later in the paper. If $b_t = 1$ then we permit a statistical model of the new hypotheses, i.e. the possible locations of the kidnapped robot. The new hypotheses born at time $t$ are characterized by a Poisson random finite set $\mathfrak{B}_t$ with intensity

$$\beta_t = \sum_{i=1}^{J_t^\beta} w_t^{(\beta,i)} \mathcal{N}(\mathbf{x};\ \mathbf{m}_t^{(\beta,i)}, \mathbf{P}_t^{(\beta,i)}) \tag{5}$$

which can approximate any arbitrary intensity function as closely as desired in the sense of the $L^1$ error [18]. The test for switching $b_t \in \{0,1\}$ will be detailed later in the paper.

Using finite set statistics (FISST), we can find an explicit expression for the random set state transition density[2]. Now the multiple-hypothesis transition density $f_{t|t-1}(\mathfrak{X}_t|\mathfrak{X}_{t-1})$ under the model (3) and with $b_t = 1$ is given by

$$f_{t|t-1}(\mathfrak{X}_t|\mathfrak{X}_{t-1}) = \prod_{\mathbf{x}\in\mathfrak{X}_t} \beta_t \cdot \prod_{\mathbf{x}_t^i \in \mathfrak{X}_{t-1}} (1 - p_S^i) \cdot$$
$$e^{-\left(\sum_{i=1}^{J_t^\beta} w_t^{(\beta,i)}\right)} \cdot \sum_\theta \prod_{i:\theta(i)>0} \frac{p_S^i f_{t|t-1}(\mathbf{x}_t^{\theta(i)}|\mathbf{x}_{t-1}^i)}{\beta_t(1-p_S^i)} \tag{6}$$

[2]In the case of random finite sets we make no distinction between the random sets and their realizations.

and with $b_t = 0$ it is given by

$$f_{t|t-1}(\mathfrak{X}_t|\mathfrak{X}_{t-1}) = \prod_{\mathbf{x}_t^i \in \mathfrak{X}_{t-1}} (1 - p_S^i) \cdot$$
$$\sum_\theta \prod_{i:\theta(i)>0} \frac{p_S^i f_{t|t-1}(\mathbf{x}_t^{\theta(i)}|\mathbf{x}_{t-1}^i)}{\beta_t(1-p_S^i)} \tag{7}$$

where the summation is taken over all associations $\theta : \{1,\ldots,N_{t-1}\} \to \{1,\ldots,N_t\}$; see [7]. To the best of our knowledge, this is the first complete set-based transition density function proposed for global robot localization.

## 2.2 The Measurement Set Model

We consider $n_s$ information sources, e.g. sensors on the robot. Each information source $j$ generates an output

$$\mathbf{Z}_t^{(j,i)} = \zeta_t^j\left(\mathbf{r}_t, \mathcal{F}_{\phi^j(i)}\right) + \mathbf{V}_t^j, \ \text{ for } i = 1,\ldots,M_t^j \tag{8}$$

in the observation space $\mathcal{M} \subseteq \mathbb{R}^{n_{z_j}}$ where typically $n_{z_j} \leq n_r$. Note that certain measurement spaces, such as bearing measurements, can be approximated by subspaces of the real line. The input $\mathcal{F}_{\phi(i)} \in \mathcal{G}$ is some feature in the global environment model $\mathcal{G}$ and $M_t^j$ is the number of measurement hypotheses generated by the $j^{th}$ source given the true robot pose $\mathbf{r}_t$ and $\mathcal{G}$. The function $\phi^j : \{1,\ldots,M_t^j\} \to \{1,\ldots,\text{number of features}\}$ relates the index of the generated measurement hypotheses to the set of features in the global model $\mathcal{G}$[3]. The input $\{\mathbf{V}_t^j\}$ is a sequence of independent Gaussian random variables. Of course, more general noise models can also be considered in this framework.

The measurement likelihood function is

$$g_t^{(j,i)}(\mathbf{z}_t^{(j,i)}|\mathbf{r}_t, \mathcal{F}_{\phi^j(i)}) = \mathcal{N}(\mathbf{z}_t^{(j,i)};\ \zeta_t^j\left(\mathbf{r}_t, \mathcal{F}_{\phi^j(i)}\right), \mathbf{\Lambda}_t^j) \tag{9}$$

where $\mathbf{\Lambda}_t^j$ is the covariance of $\mathbf{V}_t^j$.

The considered measurement model incorporates measurements of the true robot pose (or nonlinear functions of such) and false measurement hypotheses generated by ambiguities (e.g. multiple occurrences of particular features) in the environment. In addition, we account for spurious false (clutter) measurements which are caused by false detections (e.g. of unmodeled features in the environment or simply detection/recognition errors). Finally, we also consider the possibility of missed measurements.

The probability that some modeled feature $\mathcal{F}_{\phi^j(i)}$ is actually detected by sensor $j$ is given by the detection probability $p_D^j$, i.e. the probability of missing a measurement is $1 - p_D^j$. Spurious false (clutter) measurements at sensor $j$ are approximated by a Poisson random finite set $\mathfrak{C}_t^j$ with intensity

$$\kappa_t^j = \gamma_t^j \mathcal{U}(\mathcal{G}) \tag{10}$$

[3]For example, the robot might, using sensor $j$, measure the bearing to some detected feature, e.g. a door, in the environment. This single measurement constrains the robot position to a number of rays associated with this and similar features, e.g. other doors, in the environment. Each constraint is treated as a measurement hypothesis and the total number of such hypotheses is $M_t^j$.

where $\mathcal{U}(\mathcal{G})$ denotes a uniform density function over the environment. The clutter corresponds to the spurious set of measurement hypotheses generated by erroneous detections or the detection of features not in the environment model. The detection probability $p_D^j$ can be a function of the environment model $\mathcal{G}$ and the true robot pose. Thus, we can use $p_D^j$ to model the sensor geometry etc [19].

Now it follows that the set of *measurement hypotheses* at sensor $j$ is given by

$$\mathfrak{Z}_t^j = \left[ \bigcup_{i=1,\ldots,h_j} \mathfrak{D}_t(\mathbf{R}_t, \mathcal{F}_{\phi^j(i)}) \right] \cup \mathfrak{C}_t^j \quad (11)$$

where

$$\mathfrak{D}_t(\mathbf{R}_t, \mathcal{F}_{\phi^j(i)}) = \begin{cases} \{\mathbf{Z}_t^{(j,i)}\}_{i=1}^{h_j} & \text{with prob. } p_D^j \\ \emptyset & \text{with prob. } 1 - p_D^j \end{cases} \quad (12)$$

and where $\mathbf{Z}_t^{(j,i)}$ is modeled by (8) and (9) with $g_t^{(j,i)}$. Now the entire set of evidence at time $t$ is given by

$$\mathfrak{Z}_t = \bigcup_{i=1}^{n_s} \{\mathfrak{Z}_t^i, i\} \quad (13)$$

where the union is disjoint and $M_t = \sum_{i=1}^{n_s} M_t^i$. The measurement likelihood function corresponding to the single-sensor model (11) is given by

$$g_t^j(\mathfrak{Z}_t^j|\mathfrak{X}_t) = e^{-\gamma_t} \cdot \prod_{\mathbf{z}\in\mathfrak{Z}_t^j} \kappa_t \cdot \prod_{\mathbf{z}\in\mathfrak{Z}_t^j} (1 - p_D) \cdot \sum_{\theta^j} \prod_{i:\theta^j(i)>0} \frac{p_D g_t(\mathbf{z}_t(\theta^j(i))|\mathbf{x}_t^i)}{\kappa_t(1-p_D)} \quad (14)$$

where the summation is taken over all associations $\theta^j : \{1,\ldots,N_t\} \to \{1,\ldots,M_t^j\}$ and where $\mathbf{z}_t(\theta^j(i))$ is an element in $\mathfrak{Z}_t^j$ marked by the function $\theta^j$; see [7]. The multi-sensor likelihood function $g_t(\mathfrak{Z}_t|\mathfrak{X}_t)$ is then given by

$$g_t(\mathfrak{Z}_t|\mathfrak{X}_t) = \prod_{j=1}^{n_s} g_t^j(\mathfrak{Z}_t^j|\mathfrak{X}_t) \quad (15)$$

under the assumptions adopted in (11). To the best of the author's knowledge, this measurement model is the most general considered in the literature on global robot localization.

# 3 A General Bayesian Localization Algorithm

The aim of global localization is to use the measured data $\mathfrak{Z}_t$ and some dynamical constraint on the random robot pose $\mathbf{R}_t$ to estimate the set $\mathfrak{X}_t$ of potential robot positions in the environment. If $|\mathfrak{X}_t| = 1$ then the robot is said to be uniquely localized and it is of course the hope that in such cases $\mathbf{X}_t^1 \approx \mathbf{R}_t$ for the single estimate $\mathbf{X}_t^1 \in \mathfrak{X}_t$. The notion of a "set" of information points $\mathfrak{Z}_t$ and a "set" of hypotheses $\mathfrak{X}_t$ is critical as it allows us to side-step the problem of associating measurement points to a priori hypotheses in addition to other bookkeeping localization tasks.

Let $p_t(\mathfrak{X}_t|\mathfrak{Z}_{1:t})$ denote the multiple hypothesis posterior density. Then, the optimal Bayes localization filter propagates the posterior in time via the recursion

$$p_{t|t-1}(\mathfrak{X}_t|\mathfrak{Z}_{1:t-1}) = \int f_{t|t-1}(\mathfrak{X}_t|\mathfrak{X}) p_{t-1}(\mathfrak{X}|\mathfrak{Z}_{1:t-1})\, \mu_S(\mathrm{d}\mathfrak{X}) \quad (16)$$

$$p_t(\mathfrak{X}_t|\mathfrak{Z}_{1:t}) = \frac{g_t(\mathfrak{Z}_t|\mathfrak{X}_t) p_{t|t-1}(\mathfrak{X}_t|\mathfrak{Z}_{1:t-1})}{\int g_t(\mathfrak{Z}_t|\mathfrak{X}_t) p_{t|t-1}(\mathfrak{X}_t|\mathfrak{Z}_{1:t-1})\, \mu_S(\mathrm{d}\mathfrak{X})} \quad (17)$$

where $\mu_S$ is an appropriate reference measure on the collection of finite sets of $\mathcal{E}$. FISST is the first systematic approach to multi-object filtering that uses random finite sets in the Bayesian framework presented above [7, 15]. The general recursive Bayesian filter based on density functions defined for random finite set models suffers from a severe computational requirement and only a few implementations have been studied (using Monte-Carlo methods and for the problem of multi-sensor/multi-target tracking) [7, 11, 20, 21].

# 4 A First-Order Moment Approximation: The PHD Filter

The probability hypothesis density filter is an approximation developed to alleviate the computational intractability of the general Bayes filter. The PHD filter propagates the posterior intensity, a first-order statistical moment of the posterior state.

**Assumption 1.** *The predicted multi-target random finite set governed by $p_{t|t-1}$ is Poisson.*

For a random finite set $\mathfrak{X}$ on $\mathcal{E}$ with probability distribution P, its first-order moment is a non-negative function $v$ on $\mathcal{E}$, called the intensity, such that for each region $\mathcal{A} \subseteq \mathcal{E}$

$$\int |\mathfrak{X} \cap \mathcal{A}|\, \mathrm{P}(\mathrm{d}\mathfrak{X}) = \int_{\mathcal{A}} v(\mathbf{x})\, \mathrm{d}\mathbf{x} \quad (18)$$

where

$$\mathrm{E}[N] = \int_{\mathcal{E}} v(\mathbf{x})\, \mathrm{d}\mathbf{x} \quad (19)$$

and $\mathrm{E}[N]$ denotes the expected number of elements in $\mathfrak{X}$. The local maxima of $v$ are points in $\mathfrak{X}$ with the highest local concentration of expected number of elements, and hence can be used to generate estimates for the elements of $\mathfrak{X}$.

Let $v_t$ and $v_{t|t-1}$ denote the intensities associated with the multiple target posterior density $p_t$ and the multiple target predicted density $p_{t|t-1}$. The posterior intensity is $v_t(\mathbf{x}) = v_t^{n_s}(\mathbf{x})$ where

$$v_t^k(\mathbf{x}) = (1 - p_D^k) v_t^{k-1}(\mathbf{x}) + \sum_{\mathbf{z}\in\mathfrak{Z}_t^k} \frac{p_D^k g_t^{(k,')}(\mathbf{z}|\mathbf{x}) v_t^{k-1}(\mathbf{x})}{\kappa_t^k + p_D^k \int g_t^{(k,')}(\mathbf{z}|\mathbf{x}') v_t^{k-1}(\mathbf{x}')\, \mathrm{d}\mathbf{x}'} \quad (20)$$

and $v_t^0(\mathbf{x}) = v_{t|t-1}(\mathbf{x})$. The PHD predictor is given by

$$v_{t|t-1}(\mathbf{x}) = b_t\beta_t + p'_S \int f_{t|t-1}(\mathbf{x}|\mathbf{x}')v_{t-1}(\mathbf{x}')\,\mathrm{d}\mathbf{x}' \quad (21)$$

Following [7] we note that the PHD filter (similarly to the full recursive multi-target Bayesian estimator), admits explicit statistical models for missed detections, false alarms and the geometry of the sensor's field of view. In addition, the PHD filter admits explicit statistical models of robot hypothesis disappearance (death) and appearance (due to, for example, kidnapping). In addition, at every step the PHD filter computes an estimate of the number of robot hypotheses consistent with the data up until this step.

The last property aids in clutter-rejection and will be used to derive a probabilistically justified test for kidnapping.

# 5 Multi-Sensor and Multi-Hypothesis Gaussian-Sum PHD Filter

In this section we present an implementation of the PHD filer based on a mixture of Gaussians algorithm.

We firstly suppose that each hypothesis is constrained by a linear model of the form

$$\mathbf{X}_t^i = \mathbf{\Phi}_t\mathbf{x}_{t-1}^i + \mathbf{W}_t \quad (22)$$

Also, the output of information source (sensor) $j$ obeys

$$\mathbf{Z}_t^{(j,i)} = \mathbf{\Gamma}_t^j\mathbf{r}_t + \mathbf{V}_t^j \quad (23)$$

In addition, we make a reasonable assumption that the survival probability $p_S^i$ is independent of the individual state hypothesis, i.e. $p_S^i = p_S$. Under the above assumptions the following Gaussian-Mixture PHD filter (GM-PHD) is an exact implementation of the conceptual PHD filter.

**Proposition 1** ( [15])**.** *Suppose the modeling assumptions presented hold and that the posterior intensity at time $t-1$ is a Gaussian mixture of the form*

$$v_{t-1}(\mathbf{x}) = \sum_{i=1}^{J_{t-1}} w_{t-1}^i\mathcal{N}(\mathbf{x};\ \mathbf{m}_{t-1}^i, \mathbf{P}_{t-1}^i) \quad (24)$$

*Then, the predicted intensity at time $t$ is given by*

$$v_{t|t-1}(\mathbf{x}) = \beta_t + p_S\sum_{i=1}^{J_{t-1}} w_{t-1}^i\mathcal{N}(\mathbf{x};\ \mathbf{m}_{t|t-1}^i, \mathbf{P}_{t|t-1}^i) \quad (25)$$

*where*

$$\mathbf{m}_{t|t-1}^i = \mathbf{\Phi}_t\mathbf{m}_{t-1}^i \quad (26)$$

$$\mathbf{P}_{t|t-1}^i = \mathbf{\Sigma}_t + \mathbf{\Phi}_t\mathbf{P}_{t-1}^i\mathbf{\Phi}_t^\top \quad (27)$$

*and is also a Gaussian mixture.*

**Proposition 2** (Adapted from [15])**.** *Suppose the modeling assumptions presented hold and that the predicted intensity at time $t$ is a Gaussian mixture of the form*

$$v_{t|t-1}(\mathbf{x}) = \sum_{i=1}^{J_{t|t-1}} w_{t|t-1}^i\mathcal{N}(\mathbf{x};\ \mathbf{m}_{t|t-1}^i, \mathbf{P}_{t|t-1}^i) \quad (28)$$

*Then, the posterior intensity at $t$ is $v_t(\mathbf{x}) = v_t^{n_s}(\mathbf{x})$ where*

$$v_t^k(\mathbf{x}) = (1-p_D)v_t^{k-1}(\mathbf{x}) + \sum_{\mathbf{z}\in\mathfrak{Z}_t^k}\sum_{i=1}^{J_{t|t-1}} w_{t|t}^{(i,k)}(\mathbf{z})\mathcal{N}(\mathbf{x};\ \mathbf{m}_{t|t}^{(i,k)}(\mathbf{z}), \mathbf{P}_{t|t}^{(i,k)}) \quad (29)$$

$$= \sum_{i=1}^{J_t^k} w_t^{(i,k)}\mathcal{N}(\mathbf{x};\ \mathbf{m}_t^{(i,k)}, \mathbf{P}_t^{(i,k)}) \quad (30)$$

*and $v_t^0(\mathbf{x}) = v_{t|t-1}(\mathbf{x})$ and where*

$$w_{t|t}^{(i,k)}(\mathbf{z}) = \frac{p_D w_t^{(i,k-1)}q_t^{(i,k)}(\mathbf{z})}{\kappa_t + p_D\sum_{\ell=1}^{J_{t|t-1}^k} w_t^{(\ell,k-1)}q_t^{(\ell,k)}(\mathbf{z})} \quad (31)$$

$$q_t^{(i,k)}(\mathbf{z}) = \mathcal{N}(\mathbf{z}; \mathbf{\Gamma}_t\mathbf{m}_t^{(i,k-1)}, \mathbf{\Lambda}_t + \mathbf{\Gamma}_t\mathbf{P}_t^{(i,k-1)}\mathbf{\Gamma}_t^\top)$$

$$\mathbf{m}_{t|t}^i(\mathbf{z}) = \mathbf{m}_t^{(i,k-1)} + \mathbf{K}_t^{(i,k)}(\mathbf{z} - \mathbf{\Gamma}_t\mathbf{m}_t^{(i,k-1)}) \quad (32)$$

$$\mathbf{K}_t^{(i,k)} = \mathbf{P}_t^{(i,k-1)}\mathbf{\Gamma}_t^\top(\mathbf{\Lambda}_t + \mathbf{\Gamma}_t\mathbf{P}_t^{(i,k-1)}\mathbf{\Gamma}_t^\top)^{-1}$$

$$\mathbf{P}_{t|t}^i = (\mathbf{I} - \mathbf{K}_t^{(i,k)}\mathbf{\Gamma}_t)\mathbf{P}_t^{(i,k-1)} \quad (33)$$

*and $v_t(\mathbf{x})$ is also a Gaussian mixture.*

The preceding propositions show how the Gaussian components of the posterior intensity are analytically propagated in time (for the linear Gaussian measurement and hypothesis dynamic model[4]) [15].

## 5.1 Accounting for Nonlinear Models

Instead of (22) and (23) we know each state pose hypothesis $\mathbf{X}_t^i \in \mathfrak{X}_t$ is constrained by the transition density

$$f_{t|t-1}^i(\mathbf{x}_t^i|\mathbf{x}_{t-1}^i) = \mathcal{N}(\mathbf{x}_t^i;\ \psi_t\left(\mathbf{x}_{t-1}^i\right), \mathbf{\Sigma}_t) \quad (34)$$

and each measurement hypothesis likelihood function is

$$g_t^{(j,i)}(\mathbf{z}_t^{(j,i)}|\mathbf{r}_t, \mathcal{F}_{\phi^j(i)}) = \mathcal{N}(\mathbf{z}_t^{(j,i)};\ \zeta_t^j\left(\mathbf{r}_t, \mathcal{F}_{\phi^j(i)}\right), \mathbf{\Lambda}_t^j) \quad (35)$$

It then follows that the transition density for the individual hypotheses is now approximately given by

$$f_{t|t-1}^i(\mathbf{x}_t^i|\mathbf{x}_{t-1}^i) = \mathcal{N}(\mathbf{x}_t^i;\ \mathbf{\Phi}_t\mathbf{x}_{t-1}^i, \mathbf{\Sigma}_t) \quad (36)$$

where $\mathbf{\Sigma}_t$ is the covariance $\mathbf{W}_t$ and now

$$\mathbf{\Phi}_t = \left.\frac{\partial\psi_t\left(\mathbf{x}_{t-1}\right)}{\partial\mathbf{x}_{t-1}}\right|_{\mathbf{x}_{t-1}=\mathbf{m}_{t-1}^i} \quad (37)$$

In addition, the measurement likelihood functions can be approximated by

$$g_t^j(\mathbf{z}_t^j|\mathbf{x}_t) = \mathcal{N}(\mathbf{z}_t^j;\ \mathbf{\Gamma}_t^j\mathbf{x}_t, \mathbf{\Lambda}_t^j) \quad (38)$$

where

$$\mathbf{\Gamma}_t^j = \left.\frac{\partial\zeta_t^j\left(\mathbf{x}_t, \mathbf{0}\right)}{\partial\mathbf{x}_t}\right|_{\mathbf{x}_t=\mathbf{m}_{t|t-1}^i} \quad (39)$$

Now plugging the Jacobians $\mathbf{\Phi}_t^j$ and $\mathbf{\Gamma}_t^j$ into the previously outlined GM-PHD filter (covariance formulas) leads to an approximation in the spirit of the extended Kalman filter[5].

[4] In order to be computationally feasible the authors of [15] proposed a simple pruning and merging strategy.

[5] In [15] an unscented extension of the GM-PHD filter is also outlined in a similar manner.

### 5.2 The Expected Number of Hypotheses

The predicted number of hypotheses is given by

$$\mathrm{E}[N_{t|t-1}] = p_S \mathrm{E}[N_t] + b_t \sum_{i=1}^{J_t^\beta} w_t^{(\beta,i)} \tag{40}$$

while the updated, expected, number of hypotheses is given by $\mathrm{E}[N_t] = \mathrm{E}[N_t^{n_s}]$ where

$$\mathrm{E}[N_t^k] = (1 - p_D^k)\mathrm{E}[N_t^{k-1}] + \sum_{\mathbf{z}\in\mathfrak{Z}_t^k} \sum_{i=1}^{J_{t|t-1}} w_{t|t}^{(i,k)}(\mathbf{z}) \tag{41}$$

and $\mathrm{E}[N_t^0] = \mathrm{E}[N_{t|t-1}]$. Note that while some other approaches can potentially provide an estimate on the number of consistent robot poses, e.g. by counting particle clusters or hypotheses above some weight, the PHD framework inherently provides a statistically relevant estimate of the *expected* number of hypotheses consistent with the data.

## 6 Recovery from Localization Error and Kidnapping

The kidnapped robot switch $b_t \in \{0, 1\}$ allows us to switch a statistical model of new hypothesis appearance into the state hypothesis set transition model when we suspect the need to permit new hypotheses. The switch is given by

$$b_t = \begin{cases} 0 & \text{if } \mathrm{E}[N_t] \geq \tau_k \\ 1 & \text{if } \mathrm{E}[N_t] < \tau_k \end{cases} \tag{42}$$

where $\tau_k << 1$ is a kidnapping threshold on the expected number of state pose hypotheses. The idea behind the kidnapping switch is that if the expected number of hypotheses falls below some threshold (typically $<< 1$) then the robot has either been kidnapped or the localization algorithm has encountered an error. In either case, we are justified in suspecting that the new hypotheses are required (assuming we know that the true robot is still located in the environment).

The spatial distribution of the Poisson random finite birth set $\mathfrak{B}_t$ is a sum of Gaussian components $\beta_t$ and can be tailored to account for any a priori information available or can be used to approximate a uniform density over the environment model (as closely as desired [18]).

## 7 Experiments

We evaluated the localization algorithm using both simulated and real feature measurements[6]. In both cases, we use a real robot trajectory[7]. The floor plan with corner and door features is shown in Figure 2.

[6]The experiments conducted were designed to highlight a number advantages of the PHD filter framework such as providing an estimate on the expected number of robot poses and kidnap detection using real data.

[7]The robot employed in the experiments is a Pioneer P3X.

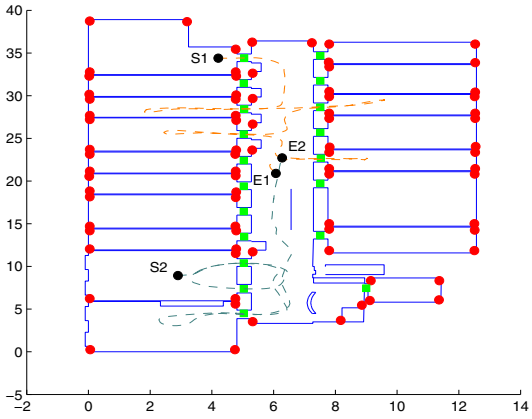



Figure 2: The floorplan at the Center for Autonomous Systems (CAS) at KTH with corner and door features highlighted. Two real robot trajectories are shown with start positions (S1 and S2) and end positions (E1 and E2).

### 7.1 Example 1

As previously stated, one particular advantage of the PHD framework is the inherent estimation of the number of robot poses consistent with the accumulated evidence and the system models. Firstly, we illustrate this behavior for a robot traveling along the first trajectory (S1→E1) shown in Figure 2 with real door features extracted as in [2]. In this case, we would expect that the expected number of hypotheses should not approach one for some time due to the symmetry in the features. The sequence of localization is illustrated in Figure 3 and verifies our expectations.

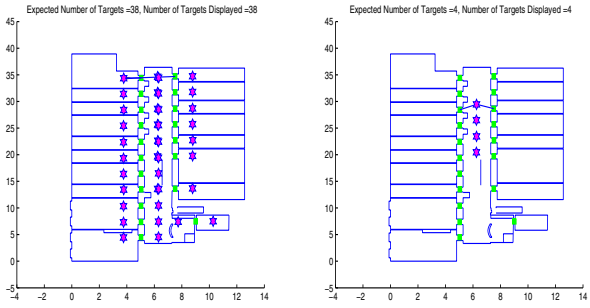
The estimated expected number of hypotheses for an individual run is depicted in Figure 4 along with the maximum Gaussian component weight scaled by this expected value.

From Figure 3 and 4 we see that the robot is uniquely localized just prior to E1 (where the expected number of hypotheses goes to $\approx 1$). We purposefully used only sparse door features to create a symmetry along trajectory 1.

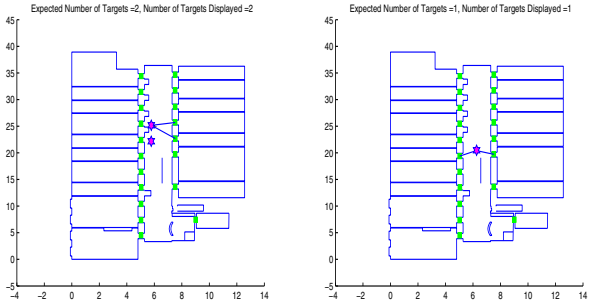
### 7.2 Example 2

In this example we highlight the re-localization capabilities of the proposed algorithm. The robot initially travels along trajectory 2 and detects only door features. At position E2 the robot is then kidnapped and placed at position S1. We expect to see the expected number of hypotheses drop to zero and the re-localization sequence should be initiated. The estimation of the expected number of hypotheses is shown in Figure 5 and verifies our expectations.

The robot is uniquely localized relatively early during the transversal of trajectory 2 (as clearly seen in Figure 5). This is because of the relatively little symmetry in the lower half of the environment model. Then following the kidnapping, we see the re-localization sequence proceed similarly to the initial localization sequence examined in Example 1. The robot is then uniquely re-localized just prior to E1.




(a) iteration=1 (b) iteration=115




(c) iteration=800 (d) iteration=1100

Figure 3: The expected number of hypotheses displayed in the title is rounded to the nearest integer value. The number of displayed hypotheses is the number of Gaussian components with a weight above 0.5. The time step between iterations is 100ms. The bearings to the measured features in the map are drawn at the true robot pose.

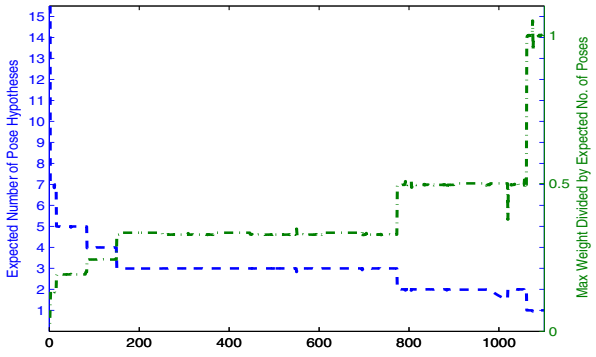



Figure 4: The expected number of hypotheses is displayed on the left for the duration of the experiment. The maximum Gaussian component weight scaled by the expected number of hypotheses is shown on the scale on the right.

We used only doors again in this example to highlight that the algorithm works on real data and to highlight again the estimation of the expected number of hypotheses (which should be greater than one initially and immediately following the first feature detection after the kidnapping detection).

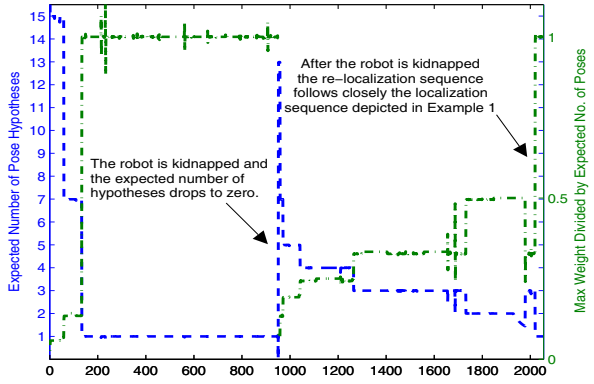



Figure 5: The expected number of hypotheses is displayed on the scale on the left. The maximum Gaussian component weight scaled by the expected number of hypotheses is shown on the scale on the right. The robot is kidnapped around iteration 940 from position E2 and placed at S1.

## 7.3 Example 3

This example is similar to Example 2 except we now use simulated door and corner features. We expect the robot to be localized much quicker (both initially and following kidnapping) as a result of the reduced symmetry.

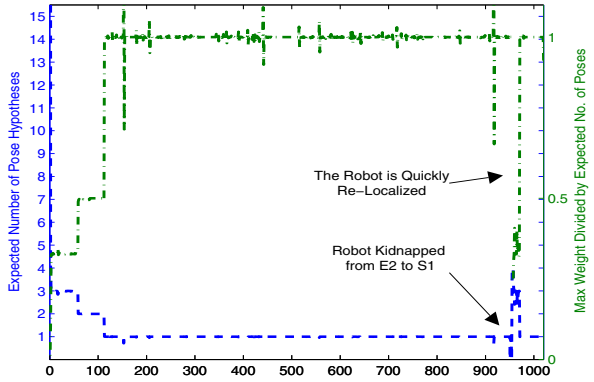



Figure 6: The expected number of hypotheses is displayed on the scale on the left. The maximum Gaussian component weight scaled by the expected number of hypotheses is shown on the scale on the right. The robot is kidnapped at some point from position E2 and placed at position S1.

Note that following re-localization, the robot follows the remainder of trajectory 1 as in the previous example (except the number of hypotheses is fewer (indeed the robot is uniquely localized) as a result of the extra corner features).

The speed at which we can detect a localization error (or robot kidnapping) depends on the chosen model parameters. We can trade robustness due to missed/false detections for an increased speed of error recovery (kidnapped detection). This trade off is inherent since in a single iteration, for ex-

ample, a kidnapped robot and a robot missing some true detections while simultaneously detecting some false positives are indistinguishable.

# 8 Conclusion

The problem of robot localization is certainly not new. However, a true recursive Bayesian solution which eliminates the data-association problem and incorporates missed/false positive detections and hypothesis birth and death etc has not been examined previously in the context of random finite sets. We have outlined the first-order moment approximation of the integrated Bayesian solution (i.e. the PHD filter) for a general class of localization problems. The algorithm provided in this paper is based on rigorous statistical analysis of random finite sets and accommodates a very general measurement and robot motion model in an integrated framework. We have presented experimental results using both real and simulated data to illustrate some of the fundamental advantages of the integrated approach.

# References


[1] S.I. Roumeliotis and G.A. Bekey. Bayesian estimation and kalman filtering: a unified framework for mobile robot localization. In *Proceedings of the 2000 IEEE International Conference on Robotics and Automation (ICRA'00)*, pages 2985–2992, 2000.

[2] Patric Jensfelt and Steen Kristensen. Active global localisation for a mobile robot using multiple hypothesis tracking. *IEEE Transactions on Robotics and Automation*, 17(5):748–760, October 2001.

[3] K.O. Arras, J.A. Castellanos, and R. Siegwart. Feature-based multi-hypothesis localization and tracking for mobile robots using geometric constraints. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA'02)*, pages 1371–1377, 2002.

[4] S. Thrun, W. Burgard, and D. Fox. *Probabilistic Robotics*. The MIT Press, Cambridge, Massachusetts, 2005. ISBN 0-262-20162-3.

[5] D. Fox, W. Burgard, F. Dellaert, and S. Thrun. Monte carlo localization: Efficient position estimation for mobile robots. In *Proceedings of the 16th AAAI Conference on Artificial Intelligence (AAAI'99).*, July 1999.

[6] S. Thrun, D. Fox, W. Burgard, and F. Dellaert. Robust monte carlo localization for mobile robots. *Artificial Intelligence*, 128(1-2):99–141, May 2001.

[7] R.P.S. Mahler. *Statistical Multisource-Multitarget Information Fusion*. Artech House, Boston, M.A., 2007.

[8] I.R. Goodman, R.P.S. Mahler, and H.T. Nguyen. *Mathematics of Data Fusion*. Kluwer Academic Publishers, London, U.K., 1997.

[9] R.P.S. Mahler. Multi-target Bayes filtering via first-order multi-target moments. *IEEE Transactions on Aerospace and Electronic Systems*, 39(4):1152–1178, 2003.

[10] H. Sidenbladh. Multi-target particle filtering for the Probability Hypothesis Density. In *Proceedings of the 2003 International Conference on Information Fusion*, pages 800–806, Cairns, Australia, 2003.

[11] M. Vihola. Random sets for multitarget tracking and data fusion. Technical report, Department of Information Technology, Tampere University of Technology, Licentiate Thesis, 2004.

[12] B.-N. Vo, S. Singh, and A. Doucet. Sequential Monte Carlo methods for multi-target filtering with random finite sets. *IEEE Transactions on Aerospace and Electronic Systems*, 41(4):12241245, 2005.

[13] A. Johansen, S. Singh, A. Doucet, and B.-N. Vo. Convergence of the sequential Monte Carlo implementation of the PHD filter. *Methodology and Computing in Applied Probability*, 8(2):265–291, 2006.

[14] D. Clark and J. Bell. Convergence results for the particle PHD filter. *IEEE Transactions on Signal Processing*, 54(7):2652–2661, July 2006.

[15] B.-N. Vo and W.K. Ma. The Gaussian mixture probability hypothesis density filter. *IEEE Transactions on Signal Processing*, 54(11):4091–4104, November 2006.

[16] D. Clark and B.-N. Vo. Convergence analysis of the Gaussian mixture probability hypothesis density filter. *IEEE Transactions on Signal Processing*, 55(4):1204–1212, April 2007.

[17] B. Kalyan, K. W. Lee, and W. S. Wijesoma. FISST-SLAM: Finite Set Statistical Approach to Simultaneous Localization and Mapping. *The International Journal of Robotics Research (In Press)*, 2009.

[18] B.D.O. Anderson and J.B. Moore. *Optimal Filtering*. Prentice Hall, Englewood Cliffs, N.J., 1979.

[19] A.N. Bishop, B. Fidan, B.D.O. Anderson, K. Dogancay, and P.N. Pathirana. Optimality analysis of sensor-target localization geometries. *Automatica*, 46(3):479–492, March 2010.

[20] H. Sidenbladh and S.-L. Wirkander. Tracking random sets of vehicles in terrain. In *Proceedings of the 2003 IEEE Workshop on Multi-Object Tracking*, Madison, Wisconsin, June 2003.

[21] W.K. Ma, B.-N. Vo, S. Singh, and A. Baddeley. Tracking an unknown time-varying number of speakers using tdoa measurements: A random finite set approach. *IEEE Transactions on Signal Processing*, 54(9):3291–3304, September 2006.

# Distributed Control of Triangular Formations with Angle-Only Constraints

Meysam Basiri, Adrian N. Bishop and Patric Jensfelt

***Abstract*—This paper considers the coupled formation control of three mobile agents moving in the plane. Each agent has only local inter-agent bearing knowledge and is required to maintain a specified angular separation relative to both neighbor agents. Assuming the desired angular separation of each agent relative to the group is feasible, then a triangle is generated. The control law is distributed and accordingly each agent can determine their own control law using only the locally measured bearings. A convergence result is established in this paper which guarantees *global asymptotic convergence* of the formation to the desired formation shape.**

## I. Introduction

This paper presents a distributed control system for triangular formation control based only on local bearing measurements and relative angular constraints. The formations considered are characterized entirely by the interior angles subtended at each agent by two neighbor agents. The angle-based formation control problem introduced in this paper is a novel contribution in the field of multi-agent dynamical systems and the control law proposed is provably globally asymptotically stabilizing.

Distributed control of multi-agent formations has been explored extensively in different settings. For example, consensus and flocking algorithms lead to formation-like steady-state structures of multi-agent systems [1]–[8]. Similarly, so-called aggregation and swarm control, which typically involves potential functions [9], is also common in the robotics and control literature [10]–[14]. A number of formation control applications have been considered [15]–[20] which typically involve formations of uninhabited aerial or underwater vehicles or formations of satellites etc. The problem considered in this paper follows closely the ideology put forth in [11], [21]–[24]. Specifically we are concerned with the formation, and subsequent maintenance, of specific inter-agent geometric relationships using distributed algorithms. The majority of existing algorithms consider only inter-agent distance measures. We differ from this in a novel way, by considering only inter-agent bearing measures taken in local coordinates, i.e. agents do not share a common heading. Our bearing-only formation control problem is motivated by the problem of optimal sensor arrangement for localization [1], [2] where the relative configurations are typically given in terms of the angular geometry.

The authors are with the Centre for Autonomous Systems (CAS) at the Royal Institute of Technology (KTH), Stockholm Sweden. This work was supported by the Swedish Foundation for Strategic Research (SSF) through CAS and also via the EU FP7 project “CogX”.

There are two fundamental problems which need addressing. Firstly, the number and characteristics of the particular constraints required has to be established. Obviously, defining a complete distance constraint graph between a group of agents will suffice in defining a unique formation. However, defining a certain (well-chosen) subset of these distance constraints can often (generically) define a unique formation, e.g. see [21], [25]–[27]. Directed constraints can also be considered, where some agents are tasked at maintaining a given distance from another agent while the converse is not true, e.g. see [26], [27]. Relative angular constraints can also be considered [28]. Establishing the constraint leads to the second problem of formation control, i.e. the design of control laws. The control laws can either be distributed or centralized. Often, distributed control lends itself naturally to the multi-agent formation control problem and it is this form of control which is considered in this paper. A distributed law for formation control is implemented by individual agents in the formation. Each agent attempts to achieve (and maintain) the desired relevant constraints placed on it’s own position but does not consider the constraints of any other agents (when planning it’s own motion control).

The contribution of this paper is the development of a distributed law for angular constrained formation control of a multi-agent system taking only relative bearing measurements. A large literature exists on bearing-only state estimation and localization [29]–[31] making the angle-based formation control problem particularly appealing. However, despite this fact, angle-based formation control is not commonly addressed in the literature; see [32], [33]. Instead, a large literature in both robotics and control focuses on distance-based formation control and potential-function-based control laws. In this paper, we introduce an angular constrained formation control problem for a group of agents tasked at maintaining a specified triangular formation. The control law introduced in this paper is globally asymptotically stabilizing given any initial agent configuration (assuming no agents are collocated initially). No similar results on provably stable angle-only formation control exist in the literature.

The paper is organized as follows. In Section II, the triangular formation control problem is introduced along with the distributed control law proposed in this paper. Subsequently, the multi-agent system evolution is examined and global stability of the desired formation shape is proved. In Section III a number of illustrative examples are given. Some discussion points are covered in Section IV and a conclusion is given in Section V.

## II. Bearing-Only Triangular Formation Control

Consider a group of $n = 3$ agents in $\mathbb{R}^2$ which *interact* via an undirected topology $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$ with $\mathcal{V} = \{1, 2, 3\}$ and $\mathcal{E} = \mathcal{V} \times \mathcal{V}$. The position of each agent is

$$\mathbf{p}_i = [x_i \ y_i]^{\mathrm{T}} \in \mathbb{R}^2 \tag{1}$$

where $x_i$ and $y_i$ denote agent $i$'s position in the $x$ and $y$ directions respectively. The neighbor set $\mathcal{N}_i \subset \mathcal{V}$ denotes the set of agents connected to agent $i$ by a single (undirected) edge. In this case $\mathcal{N}_i = \{(i+1), (i-1)\}$ (modulo $n$).

Importantly, note that agents do *not* share a common heading, i.e. they are not equipped with a compass of any kind. Agent $i$ measures only the bearing $\phi_{ij} \in [-\pi, \pi)$, $\forall j \in \mathcal{N}_i$ positive (negative) counter-clockwise (clockwise) from their local $x_i$-direction to agent $j$. Let $\alpha_i$ denote the angle subtended at agent $i$ by the two agents in $\mathcal{N}_i$. Then, the formation shape (not scale) is completely characterized by $\alpha_i$, $\forall i \in \mathcal{V}$. Introduce the following angle

$$\vartheta_i = |\phi_{i(i+1)} - \phi_{i(i-1)}| \in [0, 2\pi) \tag{2}$$

which is the angle subtended at agent $i$ by agents $i+1$ and $i-1$ which is measured positive from the $\min(\phi_{i(i+1)}, \phi_{i(i-1)})$ to $\max(\phi_{i(i+1)}, \phi_{i(i-1)})$ in agent $i$'s local coordinate frame. Then, mathematically, the interior $\alpha_i$ can be given by

$$\alpha_i = \begin{cases} \vartheta_i & \text{if } \vartheta_i \leq \pi \\ 2\pi - \vartheta_i & \text{otherwise} \end{cases} \tag{3}$$

with $\alpha_i \in [0, \pi]$. Note the difference between $\alpha_i = 0$ and $\alpha_i = \pi$ implies agent $i$ can ascertain whether or not it is in between agents $i+1$ and $i-1$ with all three collinear. Tacitly, it can be assumed that $\alpha_i$ is measured by agent $i$. The inter-agent range has not been considered and plays no part in the measurement of $\alpha_i$ or the control law to be derived.

Define the desired steady-state angles $\alpha_i^* \in [0, \pi]$, $\forall i \in \mathcal{V}$. The $\alpha_i^*$ then completely characterize the shape (not scale) of the *desired* triangle formation. The following standing assumptions are adopted to hold through Sections II and III.

**Assumption 1.** *The desired (i.e. control objective) interior angular separations $\alpha_i^*$, obey $\alpha_1^* + \alpha_2^* + \alpha_3^* = \pi$. The case where $\alpha_i^* = 0$, $\alpha_j^* \neq 0$ and $\alpha_k^* = \pi - \alpha_j^*$ is excluded.*

**Assumption 2.** *The initial agent positions $\mathbf{p}_i(0)$ are non-coincident, i.e. $\mathbf{p}_i(0) \neq \mathbf{p}_j(0)$, $\forall i \neq j$.*

Assumption 1 ensures the desired steady-state triangle is well-defined and the set of control objectives are simultaneously feasible. The case where $\alpha_i^* = 0$, $\alpha_j^* \neq 0$ and $\alpha_k^* = \pi - \alpha_j^*$ would place agent $i$ infinitely far from the other two agents and this case will be discussed separately later. The considered problem is now summarized.

**Problem** (Angle-Only Triangle Control)**.** *Design a distributed control law for agent $i$ that steers the measured angle $\alpha_i$ to $\alpha_i^*$ given any initial triangle formation. Technically, as time $t \to \infty$ then we want $\alpha_i \to \alpha_i^*$ exponentially fast given any initial configuration. Moreover, we want $\alpha_i$ to be well-defined for the entire motion of the formation, i.e. no two agent positions should coincide during the formation motion.*

This problem is novel since the controller uses only bearing measurements taken by individual agents in local coordinates and we are given only inter-agent angle constraints. Agents do not share information and agent $i$ does not consider the constraints of any other agent when executing its own control law.

### A. The Proposed Control Law

The motion of agent $i$ is governed by

$$\dot{\mathbf{p}}_i = v_i \begin{bmatrix} \cos\beta_i \\ \sin\beta_i \end{bmatrix} \tag{4}$$

where both $v_i$ and $\beta_i$ are control inputs to be determined. The heading $\beta_i$ is defined positive (negative) counter-clockwise (clockwise) from agent $i$'s local $x_i$-direction. The control law which determines $v_i$ and $\beta_i$ is truly distributed and determined solely by $\alpha_i^*$ and the measured angle $\alpha_i$ subtended at agent $i$ by two agents $j \in \mathcal{N}_i$. The speed control input of agent $i$ is defined as follows,

$$v_i = (\alpha_i^* - \alpha_i)k \tag{5}$$

where $k > 0$ is a constant (which in this paper is taken to be $k = 1$). The heading of agent $i$ is defined along the bisection of $\alpha_i \in [0, \pi]$ and toward the interior of $\alpha_i$ so that

$$\beta_i = \begin{cases} \frac{\alpha_i}{2} + \min(\phi_{i(i+1)}, \phi_{i(i-1)}), & \text{if } \vartheta_i \leq \pi \\ \frac{\alpha_i}{2} + \max(\phi_{i(i+1)}, \phi_{i(i-1)}), & \text{if } \vartheta_i > \pi \end{cases} \tag{6}$$

where $\vartheta_i$ is given by (2). Actually, it is easier to visualize the heading of agent $i$ then to mathematically define it. Visually, the heading of agent $i$ is simply toward the interior of $\alpha_i$ and specifically along the bisection of $\alpha_i$. Of course, the speed of agent $i$ might be negative. By definition, if $\alpha_i = \pi$ then the bisection is well defined by $\frac{\alpha_i}{2} + \min(\phi_{i(i+1)}, \phi_{i(i-1)})$. If $\alpha_i = 0$ then the bisection is also well defined.

The control laws (5) and (6) imply that if $\alpha_i^* > \alpha_i$, so that the angular separation subtended at agent $i$ is too small, then $v_i$ is positive and the agent moves toward the interior of and along the bisection of $\alpha_i$. Clearly, the description of agent $i$'s movement is coupled to the movements of agents $(i+1)$ and $(i-1)$.

### B. Stability Analysis for the Proposed Control Law

The range $r_{ij} = r_{ji} = \|\mathbf{p}_i - \mathbf{p}_j\|$ will be useful in analyzing the evolution of the multi-agent system but is not included in the implementation of the controller.

In addition to the formation stability results, we will show later that if $r_{ij} = r_{ji} > 0$ at some time $t_0$, for all $i, j$ then it remains strictly positive for all $t \geq t_0$, i.e. we prove that collisions are avoided naturally by our formation control law and thus $\alpha_i$ is well-defined for all time.

Consider agent $i$ with $v_i = \alpha_i^* - \alpha_i$ and heading $\beta_i$ defined as before (6) and note that $\mathcal{N}_i = \{(i+1), (i-1)\}$. Obviously, agent $i$ moves with a speed of $\alpha_i^* - \alpha_i$ and with a heading

along the bisection of $\alpha_i$. This (directly) affects how $\dot{\alpha}_{i\pm1}$ evolves. If agents $i+1$ and $i-1$ are static, then

$$\begin{aligned}\dot{\alpha}_{i+1} &= -\frac{v_i}{r_{i(i+1)}}\sin(\frac{\alpha_i}{2}) \\ &= -\frac{1}{r_{i(i+1)}}\sin(\frac{\alpha_i}{2})(\alpha_i^*-\alpha_i) \end{aligned} \quad (7)$$

using the formula for the angular velocity in terms of the cross-radial component of the velocity of agent $i$. The sign is negative since if $\alpha_i$ increases, i.e. if $(\alpha_i^*-\alpha_i)>0$, then $\alpha_{i+1}$ decreases. Similarly

$$\dot{\alpha}_{i-1} = -\frac{1}{r_{i(i-1)}}\sin(\frac{\alpha_i}{2})(\alpha_i^*-\alpha_i) \quad (8)$$

In addition, $\dot{\alpha}_i$ is affected directly by $\alpha_i^*-\alpha_i$. Note that $\sum_i \dot{\alpha}_i = 0$. Thus, when agents $i+1$ and $i-1$ are static we have

$$\begin{aligned}\dot{\alpha}_i &= \frac{(\alpha_i^*-\alpha_i)}{r_{i(i+1)}}\sin(\frac{\alpha_i}{2}) + \frac{(\alpha_i^*-\alpha_i)}{r_{i(i-1)}}\sin(\frac{\alpha_i}{2}) \\ &= \frac{r_{i(i+1)}+r_{i(i-1)}}{r_{i(i+1)}r_{i(i-1)}}\sin(\frac{\alpha_i}{2})(\alpha_i^*-\alpha_i) \\ &= \frac{\sin(\alpha_{i+1})+\sin(\alpha_{i-1})}{r_{i(i+1)}\sin(\alpha_{i+1})}\sin(\frac{\alpha_i}{2})(\alpha_i^*-\alpha_i) \\ &= \frac{\sin(\alpha_{i+1})+\sin(\alpha_{i-1})}{r_{i(i-1)}\sin(\alpha_{i-1})}\sin(\frac{\alpha_i}{2})(\alpha_i^*-\alpha_i) \end{aligned} \quad (9)$$

where the last three lines of (9) are equivalent via the sine rule. Now for future notational brevity let

$$f_{i(i+1)} = \frac{1}{r_{i(i+1)}}\sin(\frac{\alpha_{i+1}}{2}) \quad (10)$$

and let

$$g_i = \frac{r_{i(i+1)}+r_{i(i-1)}}{r_{i(i+1)}r_{i(i-1)}}\sin(\frac{\alpha_i}{2}) \quad (11)$$

where we note $g_i \geq 0$ and $f_{ij} \geq 0$ for all $i,j \in \{1,2,3\}$ when $\alpha_i \in [0,\pi]$, $\forall i$. Now, assuming all agents move with a motion governed by their individual control laws we have

$$\dot{\alpha}_i = g_i(\alpha_i^*-\alpha_i) - f_{i(i+1)}(\alpha_{i+1}^*-\alpha_{i+1}) - f_{i(i-1)}(\alpha_{i-1}^*-\alpha_{i-1}) \quad (12)$$

with $\alpha_i \in [0,\pi]$. The system of differential equations

$$\dot{\alpha} = \begin{bmatrix} -g_1 & f_{12} & f_{13} \\ f_{21} & -g_2 & f_{23} \\ f_{31} & f_{32} & -g_3 \end{bmatrix} \left( \alpha - \begin{bmatrix} \alpha_1^* \\ \alpha_2^* \\ \alpha_3^* \end{bmatrix} \right) \quad (13)$$

where

$$\alpha = \begin{bmatrix} \alpha_1 & \alpha_2 & \alpha_3 \end{bmatrix}^{\mathrm{T}} \quad (14)$$

is defined on a 2-simplex in $\alpha$-space with vertices $\alpha = [\pi\ 0\ 0]^\top$, $\alpha = [0\ \pi\ 0]^\top$ and $\alpha = [0\ 0\ \pi]^\top$. We denote this manifold by $\mathcal{M}_\alpha$.

Define the control error $e_i = (\alpha_i - \alpha_i^*) \in [-\pi,\pi]$ for each agent $i$. Then the following differential system is obtained

$$\begin{aligned}\dot{e}_i &= -\frac{\sin(\alpha_{i+1})+\sin(\alpha_{i-1})}{r_{i(i+1)}\sin(\alpha_{i+1})}\sin(\frac{\alpha_i}{2})e_i + \\ &\quad \frac{1}{r_{i(i+1)}}\sin(\frac{\alpha_{i+1}}{2})e_{i+1} + \\ &\quad \frac{1}{r_{i(i-1)}}\sin(\frac{\alpha_{i-1}}{2})e_{i-1} \end{aligned} \quad (15)$$

Using both (10) and (11), then the system of differential equations (15) can be written succinctly as

$$\dot{e}_i = -g_i e_i + f_{i(i+1)}e_{i+1} + f_{i(i-1)}e_{i-1} \quad (16)$$

Note that $\dot{e}_i$ is a nonlinear differential equation since, for example, $\alpha_i = \alpha_i^* + e_i$ is dependent on the known constant $\alpha_i^*$ and also the error $e_i$. Stacking the system of differential equations (15) or (16) leads to

$$\dot{\mathbf{e}} = \mathbf{F}(\mathbf{e})\mathbf{e} \quad (17)$$

where

$$\mathbf{e} = \begin{bmatrix} e_1 & e_2 & e_3 \end{bmatrix}^{\mathrm{T}} \quad (18)$$

and where

$$\mathbf{F}(\mathbf{e}) = \begin{bmatrix} -g_1 & f_{12} & f_{13} \\ f_{21} & -g_2 & f_{23} \\ f_{31} & f_{32} & -g_3 \end{bmatrix} \quad (19)$$

where $\mathbf{e}$ is defined on a 2-simplex in $\mathbf{e}$-space with vertices $\mathbf{e} = [\pi - \alpha_1^*\ -\alpha_2^*\ -\alpha_3^*]^\top$, $\mathbf{e} = [-\alpha_1^*\ \pi-\alpha_2^*\ -\alpha_3^*]^\top$ and $\mathbf{e} = [-\alpha_1^*\ -\alpha_2^*\ \pi-\alpha_3^*]^\top$. We denote this manifold by $\mathcal{M}_e$. In fact, $\mathcal{M}_e$ is obtained directly from $\mathcal{M}_\alpha$ via a translation by $-[\alpha_1^*\ \alpha_2^*\ \alpha_3^*]^\top$. Again, $\mathbf{F}(\mathbf{e})$ is (significantly) nonlinear in $\mathbf{e}$ since $\alpha_i = \alpha_i^* + e_i$.

Figure 1 depicts the error manifold and shows six distinct error regions, $\mathfrak{R}_{i\pm}$, with $i \in \{1,2,3\}$. The index conventions will become clear subsequently.

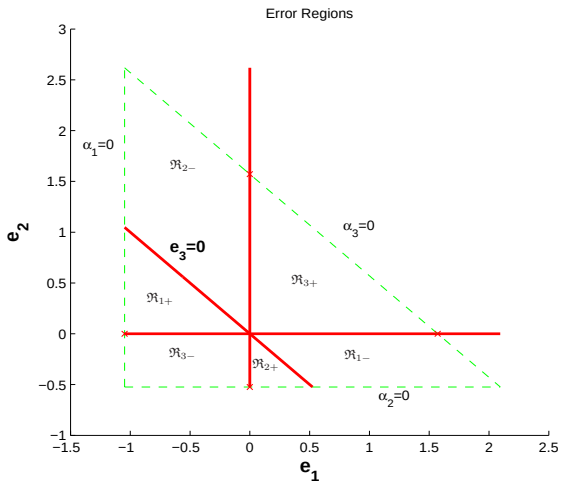



Fig. 1. A plot of the open error manifold showing six distinct regions and the boundaries of the manifold.

Note the regions are taken without boundary such that, for example, we can define $\mathfrak{R}_{3+}$ by

$$\mathbf{e} \in \mathfrak{R}_{3+} \iff \begin{bmatrix} 0 < e_1 < \pi - \alpha_1^* \\ 0 < e_2 < \pi - \alpha_2^* \\ -\alpha_3^* < e_3 < 0 \end{bmatrix} \quad (20)$$

For distinct $i,j,k \in \{1,2,3\}$, we chose the individual error regions to exhibit the following useful properties

$$\mathfrak{R}_{i+} \Rightarrow \{e_j > 0, e_k > 0, e_i < 0, \dot{e}_i > 0\} \quad (21)$$

or

$$\mathfrak{R}_{i-} \Rightarrow \{e_j < 0, e_k < 0, e_i > 0, \dot{e}_i < 0\} \quad (22)$$

where $e_i \in [-\alpha_i^*, \pi - \alpha_i^*] \subset [-\pi, \pi]$, $\forall i$ and $\sum_i e_i = 0$ must be enforced. The sign of the errors is taken directly from the definition of the region while the sign of a particular error velocity can be determined using the signs of the error and (17). *The inequalities are strict*. Note importantly that the simplex, or manifold $\mathcal{M}_e$ shifts in the error space depending on the desired configuration angles $\alpha_i^*$. As such, error regions, $\mathfrak{R}_{\mathrm{i}\pm}$, can grow or shrink, and can disappear altogether. For example, take the case where $\alpha_1^* = \alpha_2^* = 0$ such that $\alpha_3^* = \pi$, then the only region in existence is $\mathfrak{R}_{3+}$.

**Theorem 1.** *The manifold $\mathcal{M}_e$ is a positively invariant set.*

*Proof:* To show that $\mathcal{M}_e$ is positively invariant we show that for any $e_i \in \mathcal{M}_e$, then it is impossible for $e_i$ to escape $\mathcal{M}_e$. Note that $e_i \in [-\alpha_i^*, \pi - \alpha_i^*] \subset [-\pi, \pi]$. Thus, let us consider the right-sided limit,

$$\begin{aligned}
\lim_{e_i \to -\alpha_i^{*+}} \dot{e}_i &= \frac{1}{r_{i(i+1)}} \sin(\frac{\alpha_{(i+1)}}{2}) e_{i+1} + \\
&\quad \frac{1}{r_{i(i-1)}} \sin(\frac{\alpha_{(i-1)}}{2}) e_{i-1} \\
&= \frac{e_{i+1}}{r_{i(i+1)}} \quad \text{if} \quad \begin{array}{l} e_{i+1} \to (\pi - \alpha_{i+1}^*)^- \\ e_{i-1} \to -{\alpha_{i-1}^*}^+ \end{array} \\
&= \frac{e_{i-1}}{r_{i(i-1)}} \quad \text{if} \quad \begin{array}{l} e_{i+1} \to -{\alpha_{i+1}^*}^+ \\ e_{i-1} \to (\pi - \alpha_{i-1}^*)^- \end{array} \\
&> 0
\end{aligned} \tag{23}$$

which implies $e_i$ cannot escape $\mathcal{M}_e$ in one direction. A similar computation shows that $e_i$ cannot escape $\mathcal{M}_e$ in the other direction, i.e. by following $e_i \to \pi - \alpha_i^*$ through the boundary of the manifold. That is

$$\begin{aligned}
\lim_{e_i \to (\pi - \alpha_i^*)^-} \dot{e}_i &= -\frac{r_{i(i+1)} + r_{i(i-1)}}{r_{i(i+1)} r_{i(i-1)}} e_i \\
&< 0
\end{aligned} \tag{24}$$

which completes the proof. □ ■

Note that technically, once inside $\mathcal{M}_e$, there are only three possible escape routes. In the proof of Theorem 1 we show that none of these routes can be taken. Given that $\mathcal{M}_e$ is a positively invariant set, we state the following result which ensures the formation is well-defined for all time $t$, i.e. the angles $\alpha_i$ are well defined for all time.

**Theorem 2.** *Suppose that $\mathbf{p}_i(t_0) \neq \mathbf{p}_j(t_0)$ for $i \neq j$ at some time $t_0$. Then, $\mathbf{p}_i(t) \neq \mathbf{p}_j(t)$ for $i \neq j$ for all $t \geq t_0$, i.e. for all $t \geq t_0$ we have $\|\mathbf{p}_i(t) - \mathbf{p}_j(t)\| > 0$.*

*Proof:* In order for $\mathbf{p}_i(t) = \mathbf{p}_j(t)$ at some time $t > t_0$ there must exist a time interval $[t - \epsilon, t]$ with $t - \epsilon \geq t_0$ on which $\beta_i = \phi_{ij}$ and/or $\beta_j = \phi_{ji}$ for any $\epsilon \geq dt$. We now show that no such time interval can exist. We consider now, with no loss of generality, that $\beta_i = \phi_{ij}$. Note that $\beta_i = \phi_{ij}$ on $[t-\epsilon, t]$ then implies $\alpha_i = 0$ which implies $\alpha_j = 0$ or $\alpha_j = \pi$ on the entire interval $[t-\epsilon, t]$. If $\alpha_j(t-\epsilon) = \pi$ then at time $t - \epsilon + dt$ we immediately have $\beta_i \neq \phi_{ij}$ since $\alpha_j(t - \epsilon + dt) < \pi$. To see this note that $\alpha_j(t - \epsilon) = \pi$ implies

$$\dot{\alpha}_j = -g_j(\alpha_j - \alpha_j^*) \quad \text{on} \quad [t - \epsilon, t - \epsilon + dt] \tag{25}$$

which is strictly negative unless $\alpha_j^* = \pi$ which according to Assumption 1 would imply that both agents $i, k \neq j$ are also at equilibrium. Similarly, if $\alpha_j = 0$ then at time $t - \epsilon + dt$ we immediately have $\beta_i \neq \phi_{ij}$ since $\alpha_j(t - \epsilon + dt) > 0$. □ ■

The previous result ensures collisions are avoided naturally by the formation. The following result characterizes the equilibrium points of the system.

**Theorem 3.** *The system (17) is at equilibrium $\dot{\mathbf{e}} = 0$ if and only if $\mathbf{e} = 0$.*

*Proof:* The sufficiency of $\mathbf{e} = 0$ is obvious. To prove necessity, suppose firstly that the state of the system is in one of the six distinct regions $\mathfrak{R}_{\mathrm{i}+}$ or $\mathfrak{R}_{\mathrm{i}-}$ defined using (21) or (22). Using (21) or (22) it is clear $\dot{e}_i \neq 0$ for at least one $i$, i.e. the system is not at equilibrium.

Now it remains to show that on the manifold $\mathcal{M}_e$ there are no equilibrium points on the boundaries in between the error regions. Denote such a boundary via

$$\Sigma_{\mathrm{i+j-}} = \{\partial\mathfrak{R}_{\mathrm{i}+} \cap \partial\mathfrak{R}_{\mathrm{j}-}\}/\{\mathbf{0}\} = \Sigma_{\mathrm{j-i+}} \tag{26}$$

and note we consider only boundaries with strictly positive length, i.e. a strictly positive 1-d Hausdorff measure. Now following our derivation of the error regions $\mathfrak{R}_{\mathrm{i}+}$ we find that

$$\mathbf{e} \in \Sigma_{\mathrm{i+j-}} \iff \begin{bmatrix} -\alpha_i^* < e_i < 0 \\ 0 < e_j < \pi - \alpha_j^* \\ e_k = 0 \end{bmatrix} \tag{27}$$

which implies, using (16), that $\dot{e}_i > 0$ and $\dot{e}_j < 0$ and thus $\dot{\mathbf{e}} \neq 0$. This completes the proof. □ ■

We introduce the following theorem which will form the basis of our subsequent stability proof.

**Theorem 4** (Poincare-Bendixson [34])**.** *Let $\mathcal{M} \subset \mathbb{R}^2$ be a compact, positively invariant two-manifold containing a finite number of fixed points. Let $\mathbf{x} \in \mathcal{M}$ and consider the $\omega$-limit set $\omega(\mathbf{x})$. Then one of the following possibilities holds:*

1) *$\omega(\mathbf{x})$ is an equilibrium point;*
2) *$\omega(\mathbf{x})$ is a closed orbit;*
3) *$\omega(\mathbf{x})$ consists of a finite number of fixed points $\overline{\mathbf{x}}_1, \ldots, \overline{\mathbf{x}}_m$ and orbits $\gamma$ with $\alpha(\gamma) = \overline{\mathbf{x}}_i$ and $\omega(\gamma) = \overline{\mathbf{x}}_j$,*

*where $\alpha(\gamma)$ means the $\alpha$-limit set of every point $\gamma$.*

The intuition behind the Poincare-Bendixson theorem is that all bounded trajectories in a planar region (or two-manifold) must converge to an equilibrium point, a limit cycle, or a union of fixed points and the trajectories connecting them, i.e. so-called homoclinic or heteroclinic orbits.

We know there is only a single equilibrium and that $\mathcal{M}_e$ is positively invariant. We now show there are no closed orbits.

**Theorem 5.** *The system (17) has no closed orbits in $\mathcal{M}_e$.*

*Proof:* Consider the arc between adjacent regions given by

$$\Sigma_{\mathrm{i+j-}} = \{\partial\mathfrak{R}_{\mathrm{i}+} \cap \partial\mathfrak{R}_{\mathrm{j}-}\}/\{\mathbf{0}\} = \Sigma_{\mathrm{j-i+}} \tag{28}$$

with strictly positive length, i.e. a strictly positive 1-d Hausdorff measure. There are six such 'well-defined' sets $\Sigma_{\mathrm{i+j-}} =$

$\Sigma_{\mathrm{j-i+}}$. Now define

$$\Sigma = \Sigma_{3+1-} \cup \Sigma_{1-2+} \cup \Sigma_{2+3-} \cup \Sigma_{3-1+} \cup \Sigma_{1+2-} \cup \Sigma_{2-3+} \quad (29)$$

and note for clarity that $\Sigma \cap \{\mathbf{0}\} = \emptyset$. Note that any closed orbit must enclose the origin [34] and thus intersect every well-defined boundary $\Sigma_{\mathrm{i+j-}}$. As a consequence, if the origin is on a vertex of the manifold $\mathcal{M}_e$, i.e. if the desired configuration is a line formation, then obviously no closed orbits exist. Otherwise, the strategy is to show that any positive orbit $\psi^+(\mathbf{e})$ of (17) intersects $\Sigma$ in a strictly monotone sequence approaching the origin (if it intersects it in more than a point). That is, we show that if $\mathbf{e}_{m+1}$ is the $(m+1)^{\mathrm{th}}$ intersection of $\Sigma$ then $\|\mathbf{e}_{m+1}\| < \|\mathbf{e}_m\|$. Note that

$$\begin{aligned} \mathbf{e} \in \Sigma_{\mathrm{i+j-}} &\Rightarrow \dot{e}_i > 0,\ e_i < 0 \ \text{ and } \ \dot{e}_j < 0,\ e_j > 0 \\ &\Rightarrow e_k = 0 \ \text{ and } \ \|\mathbf{e}\| = |e_j| \end{aligned} \quad (30)$$

using the definition of regions where $i, j, k \in \{1, 2, 3\}$ are distinct indices. We proceed using an inductive-like argument. Suppose that $\mathbf{e}_m$ is the $m^{\mathrm{th}}$ intersection of $\Sigma$ (which also intersects $\Sigma_{\mathrm{i+j-}}$) for the positive orbit $\psi_t^+(\mathbf{e}_m)$ starting at $t = t_m$. Define a time $t_{m+1}$ and mark $\mathbf{e}_{m+1}$ as the $(m+1)^{\mathrm{th}}$ intersection of $\Sigma$ with $\mathbf{e}_{m+1} = \psi^+_{t_{m+1}}(\mathbf{e}_m)$. There exists a time $t_{m^+} \in (t_m, t_{m+1}]$ at which $\psi^+_{t_{m^+}}(\mathbf{e}_m)$ is in (i) $\Sigma_{\mathrm{i+j-}}$ or (ii) $\mathfrak{R}_{i+}$ or (iii) $\mathfrak{R}_{j-}$. We ignore the trivial case $\psi^+_{t_{m^+}}(\mathbf{e}_m) \in \{\mathbf{0}\}$ for some $t_{m^+} \in (t_m, t_{m+1})$.

(Case i): If $\psi^+_{t_{m^+}}(\mathbf{e}_m)$ is in $\Sigma_{\mathrm{i+j-}}$ then $t_{m^+} = t_{m+1}$ and $0 < e_j(t_{m+1}) < e_j(t_m)$ using (30). It follows that $\|\mathbf{e}_{m+1}\| < \|\mathbf{e}_m\|$ . We restart the argument at time $t = t_{m+1}$.

(Case ii): If $\psi^+_{t_{m^+}}(\mathbf{e}_m)$ is in $\mathfrak{R}_{i+}$ then $e_j > 0$, $e_k > 0$ and $\dot{e}_i = -\dot{e}_j - \dot{e}_k > 0$ which implies $-\dot{e}_j > \dot{e}_k$. The relevant boundaries of $\mathfrak{R}_{i+}$ are $\Sigma_{\mathrm{i+j-}}$ and $\Sigma_{\mathrm{i+k-}}$ for distinct $i, j, k \in \{1, 2, 3\}$. Now if $\mathbf{e}_{m+1} \in \Sigma_{\mathrm{i+j-}}$ then

$$\int_{t_m}^{t_{m+1}} \dot{e}_k(\tau) d\tau = 0 \ \Rightarrow \ \int_{t_m}^{t_{m+1}} \dot{e}_j(\tau) d\tau < 0 \quad (31)$$

which immediately implies $|e_j(t_{m+1})| < |e_j(t_m)|$. Using (30) it follows that $\|\mathbf{e}_{m+1}\| < \|\mathbf{e}_m\|$ and we can then restart the argument at time $t = t_{m+1}$. Now if instead $\mathbf{e}_{m+1} \in \Sigma_{\mathrm{i+k-}}$ then $-\dot{e}_j > \dot{e}_k$ implies

$$\int_{t_m}^{t_{m+1}} \dot{e}_j(\tau) d\tau = -e_j(t_m) \ \Rightarrow \ \int_{t_m}^{t_{m+1}} \dot{e}_k(\tau) d\tau < e_j(t_m) \quad (32)$$

and since $e_k(t_m) = 0$ we have $|e_k(t_{m+1})| < |e_j(t_m)|$. The consequence of this last fact is that $\|\mathbf{e}_{m+1}\| < \|\mathbf{e}_m\|$ and we can then restart the argument at time $t = t_{m+1}$.

(Case iii): If $\psi^+_{t_{m^+}}(\mathbf{e}_m)$ is in $\mathfrak{R}_{j-}$ then the argument follows similarly to that given in case (ii). □ ■

Note that Theorem 5 could be interpreted as a proof of asymptotic convergence of any solution of (17) to the origin. The following result makes this convergence precise.

**Theorem 6** (The Main Result)**.** *The equilibrium* $\mathbf{e} = 0$ *of the error system (17) is globally asymptotically stable.*

*Proof:* We use the Poincare-Bendixson theorem. Consider $\mathcal{M}_e^- = \mathrm{cl}(\mathcal{M}_e)$ where $\mathrm{cl}(\cdot)$ denotes set closure. Note that $\mathcal{M}_e^-$ is now compact with a single equilibrium and no closed orbits, via Theorems 3 and 5. Clearly, $\mathbf{e}(0)$ must be in $\mathcal{M}_e \subset \mathcal{M}_e^-$ and $\mathcal{M}_e$ acts as a positively invariant set, via Theorem 1. The Poincare-Bendixson theorem then states the $\omega$-set of any initial error in $\mathcal{M}_e^-$ contains only $\mathbf{e} = 0$. Global asymptotic stability is assured. □ ■

The previous result is our main result and concerns the global asymptotic formation stability for all desired configurations. Using a linearization argument, we can comment on the convergence rate for almost all desired formations.

**Theorem 7.** *If* $\alpha_i^* \in (0, \pi)$ *then solutions of (17) with any initial condition in* $\mathcal{M}_e$ *will converge asymptotically to the origin and there exists a neighbourhood* $\mathcal{U}$ *of the origin within which solutions converge at an exponential rate.*

*Proof:* The asymptotic stability of the origin for all desired configurations, i.e. $\alpha_i^* \in [0, \pi]$, and all initial positions follows from the main result, Theorem 6. Now note that $e_k = -e_i - e_j$ for distinct $i, j, k \in \{1, 2, 3\}$. We then reduce the dimension of (17) and obtain

$$\begin{aligned} \dot{\mathbf{e}}_{ij} &= \mathbf{F}_{ij}(\mathbf{e})\mathbf{e}_{ij} \\ \begin{bmatrix} \dot{e}_i \\ \dot{e}_j \end{bmatrix} &= \begin{bmatrix} -(g_i + f_{ik}) & (f_{ij} - f_{ik}) \\ (f_{ji} - f_{jk}) & -(g_j + f_{jk}) \end{bmatrix} \begin{bmatrix} e_i \\ e_j \end{bmatrix} \end{aligned} \quad (33)$$

with $g_i > 0$ and $f_{ij} > 0$ when $\alpha_i \in (0, \pi)$ and $g_i = f_{ji} + f_{ki}$. Linearization of (33) about the point $\mathbf{e} = 0$ leads to

$$\dot{\mathbf{e}} = \mathbf{A}_{ij}(\alpha^*)\mathbf{e} \quad (34)$$

where $\mathbf{A}_{ij}(\alpha^*)$ is a constant matrix and denotes the gradient of $\mathbf{F}_{ij}(\mathbf{e})\mathbf{e}_{ij}$ with respect to $\mathbf{e}$ and evaluated at $\mathbf{e} = 0$. Note that $\mathbf{A}_{ij}(\alpha^*) = \mathbf{F}_{ij}(\mathbf{e})|_{\alpha_i = \alpha_i^*}$. It is then easy to verify that

$$\mathrm{tr}(\mathbf{A}_{ij}(\alpha^*)) < 0 \quad (35)$$

$$\det(\mathbf{A}_{ij}(\alpha^*)) > 0 \quad (36)$$

for all $\alpha_i^* \in (0, \pi)$. Now it follows that $\mathbf{A}_{ij}(\alpha^*)$ is stable, i.e. $\mathbf{A}_{ij}(\alpha^*)$ has negative real eigenvalues, for all $\alpha_i^* \in (0, \pi)$. Now within a neighborhood of the origin $\mathcal{U}$ it follows from the Hartman-Grobman theorem [34] that solutions of (17) converge at an exponential rate when $\alpha_i^* \in (0, \pi)$.

When the desired formation is a line then linearization is inconclusive with one negative real eigenvalue and one zero eigenvalue (and additional tests would be required). □ ■

We conjecture that if the desired formation is a line then $\mathbf{e} = 0$ is also locally exponentially stable (we know it is globally asymptotically stable from Theorem 6). However, we do not explore this particular case further.

The neighborhood $\mathcal{U}$ can be made large by considering certain Lyapunov functions explicitly but the value in doing so is limited given the existence of Theorem 6. In addition, as discussed in the next subsection, we could not find a suitable Lyapunov function to show global stability. Also, the simulation results indicate an exponential convergence rate for the entire formation trajectory.

Finally, we make the following useful remark.

**Remark 1.** *Denote a formation of agents at equilibrium, i.e. with* $\alpha_i = \alpha_i^*$*, as an equilibrium formation which is defined by*

*the agent positions $\mathbf{p}_i^*$ at equilibrium. An equilibrium formation is invariant to scale, rotation and translation of the formation as a whole or reflection of any agent $i$ about the triangle edge formed by agents $i+1$ and $i-1$.*

This last remark is given for completeness and illustrates the simple fact that transforming an *equilibrium formation* in any of the referred to ways does not change the equilibrium status of the formation. However, it is of course still true that given $\mathbf{p}_i(0)$ for all $i$, the desired formation $\mathbf{p}_i^*$ is unique (given the standard uniqueness theorem [34]).

### C. Discussion on the Method of Proof

Note that we could not find a suitable Lyapunov function that would prove global stability for all desired formations given any initial configuration. In particular, testing the negative-definiteness of the time-derivative for various candidates was a significant hurdle. Variations on a number of quadratic-type candidate functions failed the negative-definite test in simulation. Indeed, $\mathbf{F}_{ij}$ in (33) is not negative definite for $\alpha_i \in [0, \pi]$. However, it was clear to us that the system evolved on a positively-invariant set and that there was only a single equilibrium. Moreover, we suspected that no limit cycles were present. As such, given the dimension of the system manifold, we know the Poincare-Bendixson theorem provides a rigorous statement concerning the asymptotic behaviour of the system trajectories. Thus, we chose to seek a globally asymptotic convergence proof through the Poincare-Bendixson theorem. An alternative route we considered was via linearization (which does lead to local exponential stability for almost all desired configurations). The disadvantage of using only linearization is that global stability does not follow (and even local exponential stability does not follow for desired line configurations using linearization alone). In any case, we believe the analysis given in this paper provides a deep insight into the nature of the proposed vector field on the manifold of interest.

## III. Examples

In this section we demonstrate the algorithm developed in this paper for distributed formation control with bearing-only measurements and relative angular constraints.

*1) Triangle to Triangle Formation:* The first example illustrates how the formation converges to an arbitrarily specified triangle (so long as the triangle is feasible) given a random initial triangle configuration. The desired triangle formation in this case is characterized by $\alpha_1^* = \pi/6$, $\alpha_2^* = \pi/4$ and $\alpha_3^* = 7\pi/12$. The formation motion is illustrated in Figure 2 along with the convergence of $|e_i|$ to zero.

The initial position of the three agents are randomly distributed in $\mathcal{M}_\alpha$ and the figure illustrates the trajectories of each agent as the formation converges upon the desired shape. This example illustrates that the control law can generate arbitrary triangle formations.

*2) Line to Triangle Formation:* Consider now the case involving three agents initially collinear. The desired formation is a triangle characterized by $\alpha_1^* = \pi/3$, $\alpha_2^* = \pi/6$ and $\alpha_3^* = \pi/2$. The formation motion is illustrated in Figure 3 along with the control error for each agent.

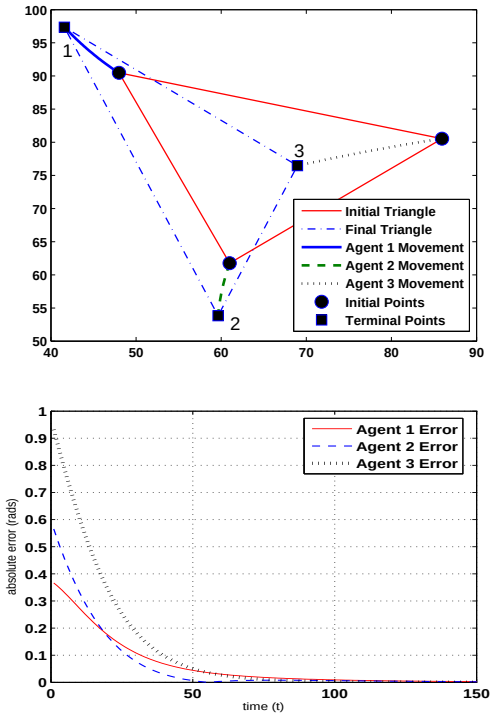



Fig. 2. The motion of the formation with a desired terminal constraint of $\alpha_1^* = \pi/6$, $\alpha_2^* = \pi/4$ and $\alpha_3^* = 7\pi/12$.

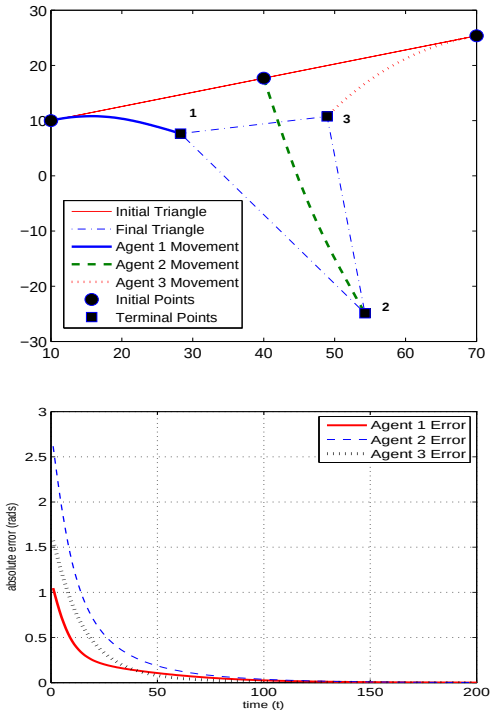



Fig. 3. The motion of a triangular formation consisting of three mobile agents initially in a collinear position with desired terminal constraints $\alpha_1^* = \pi/3$, $\alpha_2^* = \pi/6$ and $\alpha_3^* = \pi/2$.

The convergence of the three agents is illustrated in Figure 3

along with the convergence of $|e_i|$ to zero for all $i \in \{1, 2, 3\}$. This example illustrates that the control law is not affected by initial agent collinearity.

*3) Triangle to Line Formation:* This example shows the convergence of an initially random triangle formation to a desired line formation. The desired formation is characterized by $\alpha_1^* = \alpha_2^* = 0$ and $\alpha_3^* = \pi$.

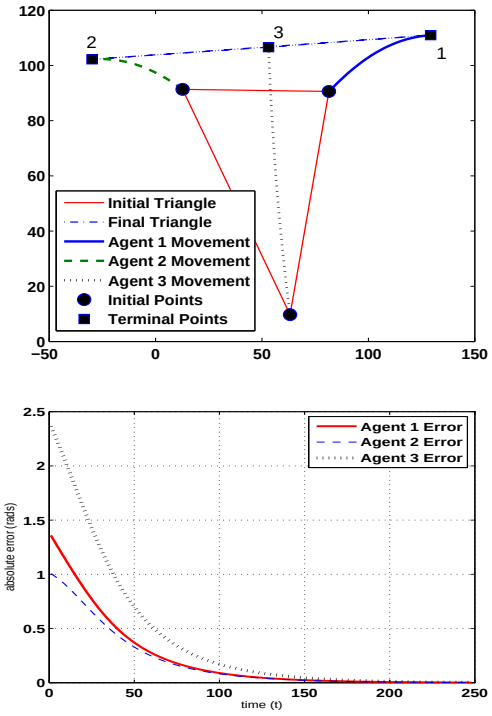



Fig. 4. The motion of a triangular formation consisting of three mobile agents starting in a random triangle and given a desired collinearity condition.

The convergence of the three agents is illustrated in Figure 4 along with the convergence of $|e_i|$ to zero for all $i \in \{1, 2, 3\}$. This example illustrates that we can steer an arbitrary initial triangle formation to a collinear formation.

*4) Line to Line Formation:* Finally, we consider the case of changing from an initial line formation with $\alpha_1 = 0$, $\alpha_2 = \pi$ and $\alpha_3 = 0$ to another (desired) line formation with $\alpha_1^* = 0$, $\alpha_2^* = 0$ and $\alpha_3^* = \pi$. The order of the agents along the line changes from the initial formation to the desired formation. The formation motion is illustrated in Figure 5 along with the control error for each agent.

The convergence of the three agents is illustrated in Figure 5 along with the convergence of $|e_i|$ to zero for all $i \in \{1, 2, 3\}$. Note that agents 2 and 3 do not collide but do indeed swap places in the formation configuration.

*5) A Phase Portrait for the System:* For illustrative purposes, we plot the phase portrait of the reduced system (33) when the desired formation is an equilateral triangle, i.e. when $\alpha_1^* = \alpha_2^* = \alpha_3^* = \pi/3$.

In Figure 6 we see the manifold $\mathcal{M}_e$ and the behaviour of the vector field on this manifold for a particular desired formation.

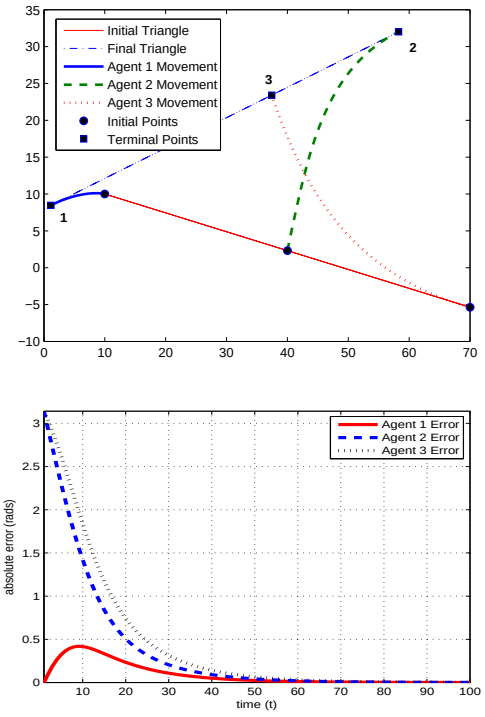



Fig. 5. The motion of a triangular formation consisting of three mobile agents initially in a collinear position with a desired condition specified by another collinear formation with a different agent ordering.

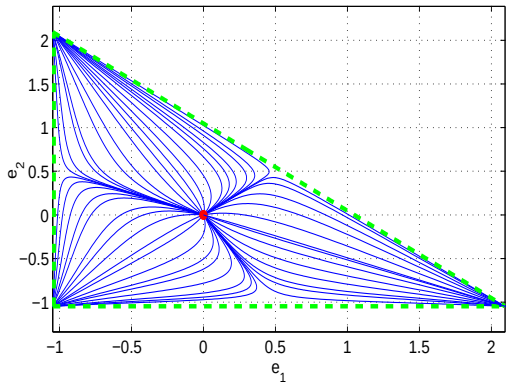



Fig. 6. Phase portrait of the reduced error system (33) when the desired formation is an equilateral triangle.

## IV. Discussion

The case where $\alpha_i^* = 0$, $\alpha_j^* \neq 0$ and $\alpha_k^* = \pi - \alpha_j^*$ is a special case where, in the desired configuration, agent $i$ must be placed infinitely far from the other two agents. Applying the derived control law in this case leads to agent $j$ and agent $k$ becoming coincident in the limit as $t \to \infty$. We note that our control law is applicable when $\alpha_i^* = \epsilon$, $\alpha_j^* \neq 0$ and $\alpha_k^* = \pi - \alpha_j^* - \epsilon$ for an arbitrarily small $\epsilon > 0$ and that inter-agent collisions are naturally avoided in such cases. In practice this is generally sufficient. We also believe an extension to the control law to account for such cases is also possible but the benefits of doing so are rather superficial.

An extension to the problem of formation control with an

arbitrary number of sensors is the next step and requires one to specify the form of the agent interaction graph which in turn specifies the constraint network for the formation. In addition, proving global stability is likely to be non-trivial as the Poincare-Bendixson theorem employed in this paper is limited to scenarios involving only three agents.

## V. Conclusion

This paper introduced a solution to the distributed bearing-only triangular formation control problem with angle-only inter-agent constraints. While the distance-based formation control problem has been extensively considered in the literature, the problem of bearing-only formation control is less studied. The solution provided in this paper requires only that each agent measure the bearing to the remaining two agents in a local coordinate system. Then, if each agent is given a desired interior angle subtended at itself by the other two agents, and assuming the set of desired interior angles is feasible, then the group of agents is shown to converge to the desired formation from any initial position.

## VI. Acknowledgement

The authors would like to thank the reviewers and editors for their insightful comments which have significantly improved the presentation of our work.

## References

[1] A.N. Bishop, B. Fidan, B.D.O. Anderson, K. Dogancay, and P.N. Pathirana. Optimality analysis of sensor-target geometries in passive localization: Part 1 - Bearing-only localization. In *Proc. of the 3rd International Conference on Intelligent Sensors, Sensor Networks, and Information Processing*, Melbourne, Australia, December 2007.

[2] A.N. Bishop, B. Fidan, B.D.O. Anderson, P.N. Pathirana, and K. Dogancay. Optimality analysis of sensor-target geometries in passive localization: Part 2 - Time-of-arrival based localization. In *Proc. of the 3rd International Conference on Intelligent Sensors, Sensor Networks, and Information Processing*, Melbourne, Australia, December 2007.

[3] A. Jadbabaie, J. Lin, and A.S. Morse. Coordination of groups of mobile autonomous agents using nearest neighbor rules. *IEEE Transactions on Automatic Control*, 48(6):9881001, June 2003.

[4] A.V. Savkin. Coordinated collective motion of groups of autonomous mobile robots: Analysis of vicsek's model. *IEEE Transactions on Automatic Control*, 49(6):981–983, June 2004.

[5] J.A. Fax and R.M. Murray. Information flow and cooperative control of vehicle formations. *IEEE Transactions on Automatic Control*, 49(9):1464–1476, September 2004.

[6] L. Moreau. Stability of multiagent systems with time-dependent communication links. *IEEE Transactions on Automatic Control*, 50(2):169–182, February 2005.

[7] R. Olfati-Saber. Flocking for multi-agent dynamic systems: Algorithms and theory. *IEEE Transactions on Automatic Control*, 51(3):401–420, March 2006.

[8] N. Moshtagh and A. Jadbabaie. Distributed geodesic control laws for flocking of nonoholonomic agents. *IEEE Transactions on Automatic Control*, 52(4):681–686, April 2007.

[9] V. Gazi. Stability analysis of swarms. *IEEE Transactions on Automatic Control*, 48(4):692–697, April 2003.

[10] E. Rimon and D.E. Koditschek. Exact robot navigation using artificial potential functions. *IEEE Transactions on Robotics and Automation*, 8(5):501–518, October 1992.

[11] R. O. Saber and R. M. Murray. Distributed cooperative control of multiple vehicle formations using structural potential functions. In *Proceedings of the 15th IFAC World Congress*, pages 1–7, Barcelona, Spain, July 2002.

[12] C. Belta and V. Kumar. Abstractions and control policies for a swarm of robots. *IEEE Transactions on Robotics*, 20(5):865–875, May 2004.

[13] Z. Lin, M.E. Broucke, and B.A. Francis. Local control strategies for groups of mobile autonomous agents. *IEEE Transactions on Automatic Control*, 49(4):622–629, April 2004.

[14] V. Gazi. Swarm aggregations using artificial potentials and sliding-mode control. *IEEE Transactions on Robotics*, 21(6):1208–1214, December 2005.

[15] V. Kumar, N.E. Leonard, and A.S. Morse (Editors). *Cooperative Control*, volume 309 of *Lecture Notes in Control and Information Sciences*. Springer-Verlag, New York, NY, 2004.

[16] W. Ren and R.W. Beard. A decentralized scheme for spacecraft formation flying via the virtual structure approach. *AIAA Journal of Guidance, Control and Dynamics*, 27(1):73–82, 2004.

[17] P. Ogren, E. Fiorelli, and N.E. Leonard. Cooperative control of mobile sensor networks: Adaptive gradient climbing in a distributed environment. *IEEE Transactions on Automatic Control*, 49(8):1292–1302, August 2004.

[18] J. Cortes, S. Martinez, T. Karatas, and F. Bullo. Coverage control for mobile sensing networks. *IEEE Transactions on Robotics*, 20(2):243255, February 2004.

[19] E. Fiorelli, N.E. Leonard, P. Bhatta, D.A. Paley, R. Bachmayer, and D.M. Fratantoni. Multi-AUV control and adaptive sampling in Monterey Bay. *IEEE Transactions on Oceanic Engineering*, 31(4):935–948, October 2006.

[20] N.E. Leonard, D.A. Paley, Lekien F., R. Sepulchre, Fratantoni D.M., and R.E. Davis. Collective motion, sensor networks and ocean sampling. *Proceedings of the IEEE*, 95(1):48–74, January 2007.

[21] R. O. Saber and R. M. Murray. Graph rigidity and distributed formation stabilization of multi-vehicle systems. In *Proceedings of the 41st IEEE Conference on Decision and Control*, pages 2965–2971, Las Vegas, Nevade, USA, 2002.

[22] J. Baillieul and A. Suri. Information patterns and hedging Brockett's theorem controlling vehicle formations. In *Proceedings of the 42nd IEEE Conference on Decision and Control*, pages 556–563, Maui, Hawaii, USA, December 2003.

[23] Z. Lin, B.A. Francis, and M. Maggiore. Necessary and sufficient graphical conditions for formation control of unicycles. *IEEE Transactions on Automatic Control*, 50(1):121–127, January 2005.

[24] B.D.O Anderson, C. Yu, S. Dasgupta, and A.S. Morse. Control of a three-coleader formation in the plane. *Systems and Control Letters*, 56:573–578, 2007.

[25] T. Eren, D.K. Goldenberg, W. Whiteley, Y.R. Yang, A.S. Morse, B.D.O. Anderson, and P.N. Belhumeur. Rigidity, computation, and randomization in network localization. In *Proceedings of the International Joint Conference of the IEEE Computer and Communications Societies*, pages 2673–2684, Hong Kong, March 2004.

[26] J.M. Hendrickx, B.D.O. Anderson, J-C. Delvenne, and V.D. Blondel. Directed graphs for the analysis of rigidity and persistence in autonomous agents systems. *International Journal of Robust and Nonlinear Control*, 17(10-11):960–981, 2006.

[27] C. Yu, J.M. Hendrickx, B. Fidan, B.D.O Anderson, and V.D. Blondel. Three and higher dimensional autonomous formations: Rigidity, persistence and structural persistence. *Automatica*, 43(3):387–402, 2007.

[28] T. Eren, W. Whiteley, and P. N. Belhumeur. Using angle of arrival (bearing) information in network localizationn. In *Proceedings of the 45th IEEE Conference on Decision and Control*, San Diego, California, USA, December 2006.

[29] A.G. Lindgren and K.F. Gong. Position and velocity estimation via bearing observations. *IEEE Transactions on Aerospace and Electronic Systems*, 14(4):564–577, July 1978.

[30] S.C. Nardone, A.G. Lindgren, and K.F. Gong. Fundamental properties and performance of conventional bearings-only target motion analysis. *IEEE Transactions on Automatic Control*, 29(9):775–787, 1984.

[31] M. Gavish and A.J. Weiss. Performance analysis of bearing-only target location algorithms. *IEEE Transactions on Aerospace and Electronic Systems*, 28(3):817–827, 1992.

[32] S.G. Loizou and V. Kumar. Biologically inspired bearing-only navigation and tracking. In *Proceedings of the 46th IEEE Conference on Decision and Control*, pages 1386–1391, New Orleans, LA, December 2007.

[33] N. Moshtagh, N. Michael, A. Jadbabaie, and K. Daniilidis. Vision-based, distributed control for motion coordination of nonholonomic robots. *IEEE Transactions on Robotics*, 25(4):851–860, August 2009.

[34] S. Wiggins. *Introduction to Applied Nonlinear Dynamical Systems and Chaos*. Springer-Verlag, New York, NY, 1990.