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

DR 3.2:
Spatial referencing and short-term vs. long-term memory

P. Jensfelt[1], H. Zender[2], C. Gretton[4], D. Skočaj[5], K. Sjöö[1], A. Aydemir[1], A. Pronobis[1], M. Göbelbacker[3], M. Hanheide[4], G.-J. Kruijff[2], J. Sotelo Chaparro[4], and P. Uršič[5]

[1] *KTH, Stockholm* [2] *DFKI GmbH, Saarbrücken* [3] *ALU, Freiburg*
[4] *BHAM, Birmingham* [5] *UL, Ljubljana*
⟨patric@kth.se⟩

Due date of deliverable:	July 31, 2011
Actual submission date:	July 15, 2011
Lead partner:	KTH
Revision:	draft
Dissemination level:	PU

This deliverable deals with qualitative spatial cognition and presents contributions in several directions. We follow up the work started last year on spatial relations by providing a perceptual model also for “in” in addition to “on”. We present results on how this can be used to analyze a scene and produce a coherent qualitative description of it. Furthermore, we continue the work on exploiting the spatial relations for the task of object search. We also incorporate planning and lift the assumption that the world is fully explored before the search is started. This allows the system to trade exploration off against exploitation. We have also completed the first full implementation of the conceptual layer of the spatial model which makes use of a probabilistic graphical model to fuse a rich set of cues to maintain probability distributions over for example room categories. The graphical model also allows us to, in a principled way, integrate long-term, generic, default knowledge with short-term, instance knowledge. The default knowledge is used for boot strapping. In another strand of work we have also looked at learning of functional spatial relations directly from sensor data in order to endow the robot with a more functional understanding of space.

Executive Summary

This report, DR 3.2, presents the work in WP3 which concerns qualitative spatial cognition during the third year in the CogX project. It follows up on the first report DR 3.1 from WP3 where the main contribution was the design of a layered model for representing space. The work on the design of the spatial model (task 3.1) has come to an end and we are now working on the implementation of it and various applications for it. More precisely we are working more on spatial relations (task 3.2), how to address the issue of short-term and long-term memory (task 3.3), how to establish references to spatial entities for human-robot interaction (task 3.4, and much related to WP6) and functional understanding of space (task 3.5). Additionally, the project wide task on representing gaps in spatial knowledge (task 3.6) is ever present.

Spatial relations were introduced during the second year as a means to perform abstraction and facilitate high-level reasoning and it has become one of the cornerstones in the work. We have continued this work by extending the repertoire of topological spatial relations from "on" to also include "in". We show in this report how these spatial relations can be used for object search but also to analyze a scene and produce a qualitative description that is appropriate for communicating knowledge about it to a human. The perceptual models for both spatial relations used are hard-coded. As a way to lift this assumption and pave the way for true functional understanding of space we have looked at methods for learning functional relations from experience by letting the robot perform experiments in simulation.

During the third year we have also continued the work on place categorization. This work was absorbed into the implementation of the conceptual map, the top layer in the spatial representation, and we now present a system that is not only able to categorize places but also maintain a wide variety of other probability distributions which it serves to the rest of the system as a basis for reasoning. The places defined in the place map are still the basic unit on which classification is performed. Spatio-temporal accumulation and integration of sensory information is done for each place. Places are clustered into rooms. When estimating the category at room level, statistics on typical indoor topologies are also incorporated. We have decoupled the low level sensory data from the high level room concepts by introducing so called properties, which characterize shape, size, general appearance trained using a corpus of low level sensor data. Training of property models is very time consuming relative to that of room concept models, which are trained using relatively little data over high-level property features. This provides better scaling and also allows, for example, new room concepts to be learned directly by having dialogues with human agents.

Object search has become our most important benchmark task in WP3 and most of the work has been driven by the challenges posed by that task.

There remain a number of open problems in object search in large spaces that cannot be solved using existing approaches to navigation. Efficient search behaviors are only conceivable if the robot has a rich understanding of the spaces in which it is operating. We make heavy use of spatial relations to achieve a hierarchical decomposition of space and allow for an implementation of indirect search. We lifted one important assumption from the first report when we removed the need for the robot to start the search with an explored map of the environment. Our robot is now able to start from only generic default knowledge and plan for a goal-direction extension of the robot's knowledge of its environment that is interleaved with object search.

Role of spatial cognition in CogX

CogX aims to produce knowledge that can help endow a robot with the ability to self-understand and self-extend. Spatial understanding is key to achieving this. Our robot shares the space with humans and is assumed to interact with them. This means that the robot's understanding must extend beyond where it is and how it can move from one place to another. It needs to be able to exchange information with people which in turn requires that the robot is able to, at least, map between its own representations of space to that of the human.

Identifying gaps in knowledge is at the heart of the CogX project. The idea is that by being able to identify these gaps the robot has come to a form of self-understanding and it can then plan to fill these gaps and thereby achieve self-extension. Examples of gaps in the context of WP3 are unknown room categories and unknown position and spatial relations between objects.

Contribution to the CogX scenarios and prototypes

The work in WP3 contributes mostly to the Dora demonstrator (WP7) by its focus on a mobile robot in large-scale spaces (beyond the current sensor horizon). With the work in analyzing table top scenes and producing qualitative descriptions of them based on the topological spatial relations defined in task 3.2, we have come closer to merging the work with that going on in the George demonstrator (WP7).

The work in WP3 forms the foundation in terms of functionality for Dora. The work on adaptive situated dialogue processing (WP6) in Dora deals with large-scale space and makes use of the representations derived in WP3. The planning system (WP4) looks at the spatial representation to find information to base its actions on. WP3 provides the container and integration mechanism for some of the knowledge gained by the visual perception system (WP2).

1 Tasks, objectives, results

1.1 Planned work

WP3 deals with qualitative spatial cognition. We have the implicit assumption that there will be processes dealing with quantitative aspects in parallel and therefore some of the work we do deals with that as well. Understanding and reasoning about space is needed for navigation; i.e. being able to move about, knowing your position. This is an example of a fundamental requirement for a mobile robot such as Dora. However, in CogX the aim is set higher and we want to, for example, investigate methods to endow the system with the capability to interact with humans in a, for the human, natural way and to be able to make use of the vast amount of knowledge that is available in various databases in human readable form. To support this, the system must be able to, among other things, make of use human concepts of space, distill sensory level information to a symbolic level, combine innate and acquired knowledge and learn over time.

We have divided the work in this work package into six tasks. One of these, dealing with the development of a spatial model (task 3.1, see Fig. 1), has come to an end and now continues in more specialized tasks (3.2-5) and one task (3.6) which represents the tasks common to the entire project, underpinning all the activities. The tasks we planned to work on the third period were:

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 3.4: Establishing reference to spatial entities for human-robot interaction. *The goal is to investigate, in the context of human-robot interaction, how the robot can refer to objects based on their spatial relations and how to learn this.*

Task 3.5: Functional understanding of space. *The goal is to investigate how to gain knowledge about the function of space by analyzing spatial models over time.*

The work in task 3.2 serves primarily two purposes, as a way to perform spatial abstraction to facilitate more efficient representations and learning and as a means for human-robot communication. This year we planned to expand our set of spatial relation from “on” to also include “in”. These two together give us a powerful basis for representing and reasoning about space.

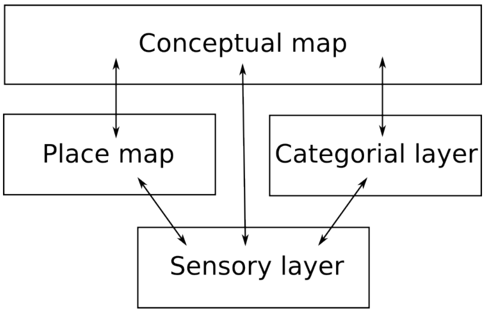


Figure 1: An overview of the four layered spatial model. The place layer and the conceptual layer have more concrete instantiations and are therefore referred to as maps.

Task 3.3 is partly fueled by the work in 3.2. The spatial relations help us structure and represent long term memory. Task 3.3 is a quite general task that comes in most of the other tasks. For any system that works with a continuous flow of data there is a need to determine what information to keep, for how long and what to do with it when its validity expires. During the previous period we looked at, for example, incremental learning. This period we will continue the work on the conceptual layer which gave the first results just before the last review. The conceptual map allows for seamless integration of long term, generic, default knowledge and more short term instance knowledge.

In task 3.4 we planned to look at concrete ways to show that the spatial relations developed in task 3.2 indeed are useful for creating representation that can support human-robot interaction. During the previous period we showed that they are useful for synthesis in an object search task (*large-scale space*), this year we will look at using them for analysis of a scene (*small-scale space*). These representations are the basis for the production and understanding of verbal references in spatially situated human-robot dialogues in WP6.

We have planned to take the first steps towards a deeper functional understanding of space in task 3.5 this year. The functional spatial relations developed in task 3.2 (”on” and ”in”) have so far been hard-coded. We planned to look at ways to make the robot learn such relations starting from more basic functional distinctions such as support and containment.

1.2 Actual work performed

In this section we describe what we actually did during the third year. The work that was started on object search during the first two years has continued and has become the task that has been used to motivate a lot of the activities in the work on the qualitative spatial cognition. It is a good task as it involves many of the elements that we want to study in CogX. For example, there is a user that the robot interacts with, the robot needs to perform the basic navigation skills, it has to build a representation of space, and it has to plan for actively gathering information to find the object as brute force search in a large environment is very costly and inefficient. As a reminder of previous work the spatial model is shown in Figure 1.

1.2.1 Task 3.2: Spatial Relations

During the second year of the project we introduced a perceptual model for the topological spatial relation “on”. This year we have extended the robot’s repertoire with “in” [31, 33] (Annex 2.1 and 2.4). These models make use of visual information and have been validated in real world experiments. They support both analysis of a perceived scene and synthesis of a scene based on information from some source such as a human or prior knowledge.

In [31] (Annex 2.1) we showed how these spatial relations can be exploited to implement an efficient search strategy based on the indirect search paradigm introduced by Garvey [10]. The idea is simple but powerful. If you want to find a small object such as a whiteboard pen it is often more efficient to look for larger objects (or more exactly objects that are easier to find) which have a strong correlation with the target object, such as a whiteboard. We made use of the spatial relations to formalize the indirect search and build chains of relations such as “the mug is IN the box ON a table (in the room)”. In this work it was assumed that the relation was given and we worked in a single room.

In [2] (Annex 2.2) we extended the work to include two rooms, a number of different relations each with an associated probability and an MDP style planner to reason about what room to search in and which of many spatial relations to exploit. Note that each long chain of relations (such as mentioned above) has to be broken down into pieces and each one taken into account by the planner. The planner also needed to trade using indirect search off against making use of all the relations in the chain at once. Consider, “the mug is ON the table IN the room”. Full indirect search would make the robot first look for the table and then the mug whereas the robot could also directly make use of the “ON the table” part of the chain and look for a mug at the corresponding height.

One of the limitations of the previous work was that it assumed that the environment was fully explored. In [1] (see DR.4.3) we lift that assumption and extend the problem to also include the tradeoff between exploring new space and exploiting, i.e. searching the part of space so far known. The dedicated MDP planner has also been replaced by a more general purpose and novel switching planner reported on in more detail in WP4 where [1] is included as well. In this work we also lift the assumption that we are dealing with specific instances of objects and therefore move from, for example, “the mug” to “a mug”. This affects the way percepts affect the probability distribution for the existence of objects.

1.2.2 Task 3.3: Short-term vs long-term spatial memory

As already mentioned we consider the work on spatial relations to be part of the work also on investigating short-term vs. long-term memory. In addition to this stream of work we have spent considerable effort on the implementation of the conceptual layer of our spatial model. This is where the high level reasoning takes place and it acts as a bridge between the robot’s representation of space and that of its human users. In this way the conceptual map plays an integral role also in task 3.4.

For the conceptual map the two benchmark tasks have been place categorization and object search. The former is included in the support needed for the latter; however, this year we have made considerable efforts to formally include objects also in the conceptual map complete with probabilistic information. This allowed us to make the conceptual map the bridge not only to the human but also to the planner so that all symbols and associated probabilities are provided to the planner from one and the same source.

Work in place categorization has up until now trained the categorical models for places directly from sensor data. This has some severe disadvantages. It is very computationally heavy and time consuming to update these models as all of them must be updated whenever a new one is added, for example. Training a new model always involves going down all the way to the sensor level. However, many concepts will be learned in the interaction with a human and this interaction will not include any exchange of sensor data. Instead it is likely that the human will describe a new room concept in terms of other concepts such as the shape and size.

In [26] and [25] (Annex 2.3 and 2.6 respectively, with 2.6 providing more details and background) the new conceptual map is presented. One of the fundamental contributions of the work for place categorization is the decoupling of the low level sensory data from the high level room concepts by introducing so called properties. These properties are shape, size, general appearance and the existence of objects. Classifiers based on low level sensory data (vision and laser) are learned for these properties. We consider these to belong to the long-term memory of the robot. The high level room concepts can now be learned based on information from, for example, common sense databases and in principle also directly from humans (yet to be demonstrated at the time of writing). In addition to requiring a lot less training data to form a new concept, significantly less memory and better scaling by reusing the low level categorical model, we can also imagine using different high level concepts for different users, which would not be possible with the old representation.

The conceptual map goes well beyond point based place categorization, however. Each node in the place map is represented in the conceptual map with associated properties. The places are connected and divided into rooms based on the detection of doors. Statistics on topological information are used as an extra source of information to influence the place categorization. As an example, this means that by having gathered strong evidence that a certain area is an office, the adjacent area will be very likely to be a corridor, as this is the most common configuration. All the information in the conceptual map is captured in a probabilistic graphical model, a chain graph. This model allows the conceptual map to answer queries not only about the category of a certain place or area but also conditional probabilities such as what the probability is to find a certain object type in a certain room or room category. This gives it a central place in the object search system and the Dora system as a whole.

To summarize, the main additions to the conceptual layer this year was an implementation based on a chain graph, the introduction of spatial properties as a way to decouple low level sensory data and high-level concepts, the use of a statistical model for the topology of space to influence the categorization, and a principled way to connect the existence and quantity of objects with other properties of space.

1.2.3 Task 3.4: Establishing reference to spatial entities for human-robot interaction

In the kinds of mobile robotics scenarios we are dealing with in CogX, more specifically in the Dora scenario, we are faced with interaction settings that need not be confined to the immediate surroundings of the human and the robot. Situated human-robot dialogues about entities (i.e., things, places, properties, or events) in large-scale space require the interlocutors to draw attention to entities that are not currently observable, and, likewise, to comprehend which remote places and things are being talked about. WP3 investigates spatial knowledge bases that are suitable for such situated communication between a robot and a human. In DR.6.4 we present an approach to producing and understanding situationally appropriate referring expressions (REs) during a discourse about large-scale space that is based on the spatial representations and knowledge bases developed within WP3 [45]. As an illustration, imagine a service robot that is supposed to clean up an apartment consisting of several rooms. The apartment contains several balls, boxes, and tables (see Figure 2). Rather than expecting an overly verbose instruction like “take the ball in the kitchen and put it into the box on the table in the kitchen”, the robot should be able to understand the more natural utterance “take the ball in the kitchen and put it into the box on the table” in the same situation. There might be different boxes that are on tables – rendering the expression “the box on the table” ambiguous with

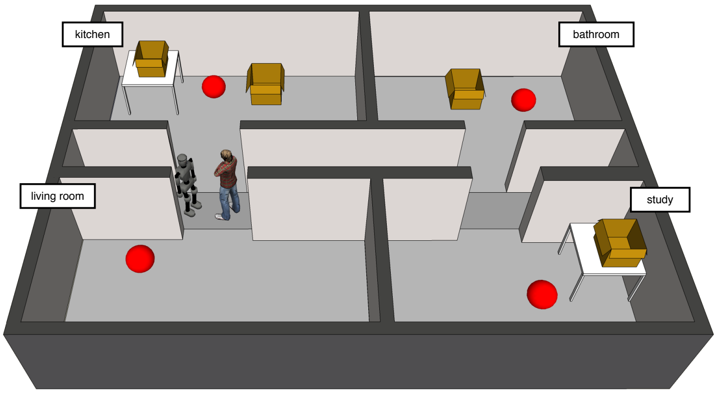


Figure 2: Four room apartment used to illustrate situationally appropriate referring expressions.

respect to the whole apartment. However, the preceding reference to "the ball in the kitchen" has shifted the focus of attention to the kitchen – which in turn allows to felicitously refer to the box on the table in the kitchen as "the box on the table.".

As a way to facilitate situated dialogue and coherent spatial reasoning also on a smaller scale, the perceptual models of "on" and "in" are complemented with a set of rules or axioms. The axioms introduce relational first-order logical predicates On(x,y) and In(x,y) in [33]. For example, the axioms introduce transitivity for "on" and "in". To do so we introduce a third relation symbol On_t (transitive On) which allows the robot to deduce that if x is on y and y is on z then x is also on z (transitively on). In total, 12 axioms are presented. Axioms and first-order logic are by nature truth functional. However, the real world and sensing of it is not crisp at all. We therefore present a way to make use of the axioms in a probabilistic framework. We do so by using factor graphs which are also the vehicle by which the chain graphs from the conceptual map in tasks 3.3 are evaluated. We implemented these on our system and showed in experiments [33] that the robot is able to analyze a scene and produce a qualitative evaluation of it, suitable for conveying information about the scene to a human user.

Another stream of research investigates learning spatial relations with a learning framework based on odKDEs that is being developed in WP5. The setting is active learning of deictic spatial relations during interaction with a tutor. This work so far falls mostly into WP5 but activities will shift gradually more into WP3 during the last year.

1.2.4 Task 3.5: Functional understanding of space

In task 3.2 where we made heavy use of topological spatial relations we made the assumptions that these and the corresponding perceptual models were given to the robot. If the robot is truly to get an understanding of space it has to be endowed with the ability to learn such relations on its own. In [32] we take one step in this direction by only providing the robot with basic functional distinctions such as support, location control, protection and confinement and let the robot learn models of these from a large number of features that can be directly perceived or calculated given a known geometry of the objects by the robot (such as positions, distances between objects, etc). The robot gathers data that it can use to learn models of interactions by performing experiments in a simulator. The simulation environment allows us to perform a large number of experiments to gather enough data and validate that it is in fact possible to learn such models.

1.2.5 RGB-D perception

In addition to the tasks that were defined almost four years ago in the project proposal, we have also identified the need to exploit the recently released affordable 3D sensor from PrimeSense/Microsoft in the context of the project. This new sensor, the so called Kinect, has created a tremendous activity focused on perception and modeling with RGB-D data (D for depth). All the partners' robots have already been equipped with a Kinect. We are collecting a RGB-D dataset with data from each site that will be released publicly. This will be one of the first datasets captured from a moving robot platform. Figure 3 shows an example of a 3D model built from part of the data from the data collected at Birmingham.

The dataset will be made available at: `http://www.cas.kth.se/rgb-d`.

1.3 Relation to state-of-the-art

In this section we briefly relate our work to the state-of-the-art. A more in depth discussion can be found in the annexes.

1.3.1 Task 3.2

Our work on quantifying spatial relations is not the first in the area. However, to our knowledge our work is the first to take a functional approach and place emphasis on being able to use the models with real sensor data. Our spatial relations “on” and “in” are based on objective properties such as support and containment. We also model and use the 3D nature of the objects in contrast to other related work. In previous and related work, the *Attention Vector Sum* was introduced in [23] as a numerical measure of

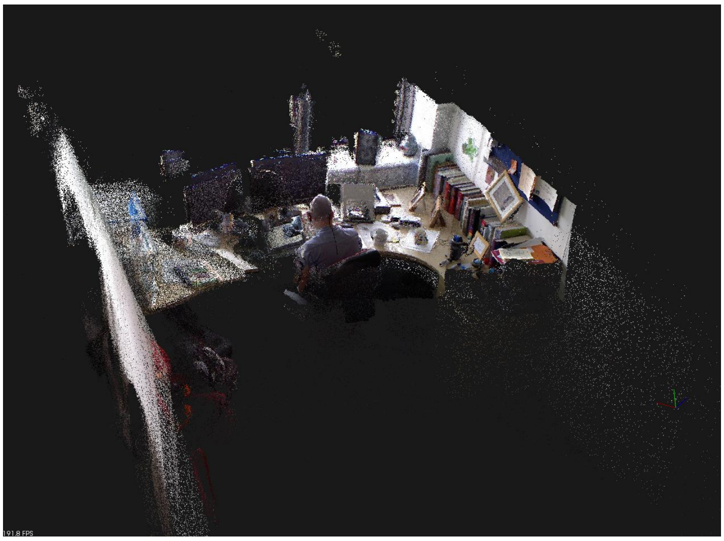


Figure 3: Local 3D model built from the data from an office at Birmingham.

how acceptable a particular spatial relation is for describing a scene. [16] proposes a model based on spatial templates describing a certain situation that can be matched in a scene. Spatial relations used in the context of interaction with users in a graphical way are presented in [15] and [12]. The latter also provides a good survey of computational models for spatial relations. We believe topological relations are key for a robot's understanding of space. These types of relations were surveyed in [4]. The *Region connection calculus* (RCC) [7] and its variants are well-known approaches providing a language for expressing qualitative relationships between regions – such as containment, tangential contact etc. RCC is purely geometrical and therefore does not involve functional relations.

1.3.2 Task 3.3

The problem of place recognition [13, 39, 36, 3, 30, 40, 28] and place categorization [37, 19, 9, 44, 42, 27] has a long history in computer vision and robotics. Most of the previous work has focused on one sensory modality (typically laser or vision) or one cue (e.g. geometry, appearance, objects) but there are several examples where vision has been combined with information from sensing of geometrical features of space [36, 20]. When used in a mapping context some work specializes on handling large-scale datasets such as in [18, 6] and therefore require extreme computational efficiency. We instead focus on a method that is well adapted to be used in a human interaction setting where the robot, in addition of building up a representation of the here and now (short-term memory) also must be able to relate its knowledge to human concepts stored in long-term memory and provide mechanism to learn such concepts effciently with little data.

Many have observed that it is important to be able to fuse information over time and space. Some have applied typically mapping techniques such as particle filters [27] and other employed Bayesian filtering [43] or graphical models such as HMM [38]. In our work we perform integration at two levels, first at the level of places where information is accumulated and later the conceptual map where information is fused across places and combined with typical room connectivity information by a chain graph [14] which is a probabilistic graphical model.

1.3.3 Task 3.4

In linguistics, referring expressions are definite descriptions (typically noun phrases) that enable a hearer to "pick out whom or what [the speaker] is talking about" [8]. In the field of Natural Language Generation the task of generating referring expressions is finding an appropriate verbal expression that successfully identifies the intended referent to the hearer on first mention. As long as the domains of discourse are small visual scenes or other closed-context scenarios, the intended referents are always in the current focus of attention. In contrast, we address the challenge of producing and understanding references to entities that are *outside* the current focus of attention, e.g., because they have not been mentioned yet and are beyond the currently observable scene; a situation that is common when interacting with a mobile indoor robot.

Paraboni *et al.* [24] are among the few to address the issue of generating references to entities outside the immediate environment. They present an algorithm for context determination in hierarchically ordered domains. Large-scale space can be viewed as a hierarchically ordered domain [35, 17].

In DR.6.4 we present an extension of our previous work on determining appropriate contexts in spatial domains [46, 47]. We advance the state-of-the-art by not only looking at single referring expressions, but rather taking into account how the focus of attention shifts through the spatial domain during a discourse about large-scale space (see Section 1.2.3). The approach lends itself to be used with the kinds of spatial knowledge bases that are investigated and developed in WP3. A more detailed account can be found in DR.6.4.

The approach to quantifying spatial relations ("in" and "on") by detecting functional relations (support and containment, resp.) is immediately applicable in the context of state-of-the-art approaches to the generation of referring expressions in shared visual scenes (see, e.g., [41]).

1.3.4 Task 3.5

There has been some activity related to learning of spatial relations before such as Regier [29] and Skočaj et al. [34]. In these works the systems learn to recognize spatial relations and associate them with words.

Our work under this task is highly related to the idea of *affordances* introduced by Gibson [11]. We want to learn affordances and therefore take the view of Norman [22] who argued that an agent must be aware of the capabilities of an object for an affordance to exist whereas affordances are independent of the knowledge or predisposition of an agent according to Gibson.

Cos-Aguilera et al. [5] and Mugan and Kuipers [21] both show examples of affordance learning in the context of recognizing objects and action rules, both in simulated environments. Our work applies those principles to the problem of learning spatial relations: functionally defined, bottom-up and carried out in simulation.

2 Annexes

In addition to the papers presented in these annexes the following two papers are highly related to the work in WP3

- M. Hanheide, C. Gretton, R. W. Dearden, N. A. Hawes, J. L. Wyatt, A. Pronobis, A. Aydemir, M. Göbelbecker, and H. Zender, "Exploiting probabilistic knowledge under uncertain sensing for efficient robot behaviour", IJCAI, Barcelona, Spain, July 2011. (in DR.7.2 Annex A.1)
- A. Aydemir, M. Göbelbecker, A. Pronobis, K. Sjöö and P. Jensfelt, "Plan-based Object Search and Exploration Using Semantic Spatial Knowledge in the Real World", to appear ECMR, Örebro, Sweden, September 2011, (in DR.4.3 Annex 2.2)

2.1 K. Sjöö et al. “Topological spatial relations for active visual search” (Tech report 2010)

Bibliography Kristoffer Sjöö, Alper Aydemir, David Schlyter and Patric Jensfelt, “Topological spatial relations for active visual search”, KTH CSC, CAS/CVAP, September 2010, TRITA-CSC-CV 2010:2 CVAP 317

Abstract If robots are to assume their long anticipated place by humanity’s side and be of help to us in our unstructured environments, we believe that adopting human-like cognitive patterns will be valuable. Such environments are the products of human preferences, activity and thought; they are imbued with semantic meaning. In this paper we investigate qualitative spatial relations with the aim of both perceiving those semantics, and of using semantics *to* perceive. More specifically, in this paper we introduce general perceptual measures for two common *topological spatial relations*, “on” and “in”, that allow a robot to evaluate object configurations, possible or actual, in terms of those relations. We also show how these spatial relations can be used as a way of guiding visual object search. We do this by providing a principled approach for *indirect search* in which the robot can make use of known or assumed spatial relations between objects, significantly increasing the efficiency of search by first looking for an intermediate object that is easier to find. We explain our design, implementation and experimental setup and provide extensive experimental results to back up our thesis.

Relation to WP This paper presented perceptual models for the topological spatial relations “in” and “on” which is one of the main ways in which we perform abstraction of spatial knowledge. Object search is used as an example task to show that this is a useful abstraction which allows us to realise so called indirect search. (Task 3.2)

2.2 A. Aydemir et al., “Search in the real world: Active visual object search based on spatial relations”

Bibliography Alper Aydemir, Kristoffer Sjöö, John Folkesson, Andrzej Pronobis, and Patric Jensfelt, “Search in the real world: Active visual object search based on spatial relations”, International Conference on Robotics and Automation (ICRA11), May 2011, Shanghai, China

Abstract Objects are integral to a robot’s understanding of space. Various tasks such as semantic mapping, pick-and-carry missions or manipulation involve interaction with objects. Previous work in the field largely builds on the assumption that the object in question starts out within the ready sensory reach of the robot. In this work we aim to relax this assumption by providing the means to perform robust and large-scale active visual object search. Presenting spatial relations that describe topological relationships between objects, we then show how to use these to create potential search actions. We introduce a method for efficiently selecting search strategies given probabilities for those relations. Finally we perform experiments to verify the feasibility of our approach.

Relation to WP This paper show how to make use of topological spatial relations and planning for efficient object search. The spatial relations allow us describe the location of objects in way that scales well to large environments and fits well with the indirect search paradigm which is realised using an MDP style planner. We make use of a both of long-term and short-term knowledge in this task. We store, for example, generic knowledge such as models for objects and the typical relations between them in long-term memory and, for example, a local geometric model describing the local surroundings in short-term memory. (Tasks 3.2 and 3.3)

2.3 A. Pronobis and P. Jensfelt, “Hierarchical Multi-Modal Place Categorization”

Bibliography Andrzej Pronobis and Patric Jensfelt, “Hierarchical Multi-Modal Place Categorization”, to appear at European Conference on Mobile Robotics, September 2011, Örebro, Sweden

Abstract In this paper we present an hierarchical approach to place categorization. Low level sensory data is processed into more abstract concept, named *properties* of space. The framework allows for fusing information from heterogeneous sensory modalities and a range of derivatives of their data. Place categories are defined based on the properties that decouples them from the low level sensory data. This gives for better scalability, both in terms of memory and computations. The probabilistic inference is performed in a chain graph which supports incremental learning of the room category models. Experimental results are presented where the shape, size and appearance of the rooms are used as properties along with the number of objects of certain classes and the topology of space.

Relation to WP This paper relates to our work on knowledge representations at different time-scales (Task 3.3) as well as as a support for interaction with humans (Task 3.4).

2.4 K. Sjöö et al., "Functional topological relations for qualitative spatial representation"

Bibliography Kristoffer Sjöö, Andrzej Pronobis and Patric Jensfelt, "Functional topological relations for qualitative spatial representation", The 15th International Conference on Advanced Robotics, June 2011, Tallin, Estonia

Abstract In this paper, a framework is proposed for representing knowledge about 3-D space in terms of the functional *support* and *containment* relationships, corresponding approximately to the prepositions "on" and "in". A perceptual model is presented which allows for appraising these qualitative relations given the geometries of objects; also, an axiomatic system for reasoning with the relations is put forward.

We implement the system on a mobile robot and show how it can use uncertain visual input to infer a coherent qualitative evaluation of a scene, in terms of these functional relations

Relation to WP This paper shows how to make use of spatial relations (Task 3.2) to build up a representation of a scene perceived with vision in a way that would allow the system to describe the scene qualitatively to a human (Task 3.4).

2.5 K. Sjöö and P. Jensfelt, "Learning spatial relations from functional simulation"

Bibliography Kristoffer Sjöö and Patric Jensfelt, "Learning spatial relations from functional simulation", to appear at IEEE/RSJ International Conference on Intelligent Robots and Systems, September 2011, San Fransisco, USA

Abstract Robots acting in complex environments need not only be aware of objects, but also of the relationships objects have with each other. This paper suggests a conceptualization of these relationships in terms of task-relevant functional distinctions, such as support, location control, protection and confinement. Being able to discern such relations in a scene will be important for robots in practical tasks; accordingly, it is demonstrated how predictive models can be trained using data from physics simulations. The resulting models are shown to be both highly predictive and intuitively reasonable.

Relation to WP This paper presents our first steps towards endowing the robot with the ability to acquire a functional understanding of space. The acquisition is done by performing mini-experiments in simulation. (Task 3.5)

2.6 A. Pronobis, "Semantic Mapping with Mobile Robots"

Bibliography Andrzej Pronobis, "Semantic Mapping with Mobile Robots", Royal Institute of Technology (KTH), SE-100 44 Stockholm, Sweden, June 2011, ISBN 978-91-7501-039-7, ISSN-1653-5723

Abstract After decades of unrealistic predictions and expectations, robots have finally escaped from industrial workplaces and made their way into our homes, offices, museums and other public spaces. These service robots are increasingly present in our environments and many believe that it is in the area of service and domestic robotics that we will see the largest growth within the next few years. In order to realize the dream of robot assistants performing human-like tasks together with humans in a seamless fashion, we need to provide them with the fundamental capability of understanding complex, dynamic and unstructured environments. More importantly, we need to enable them the sharing of our understanding of space to permit natural cooperation. To this end, this thesis addresses the problem of building internal representations of space for artificial mobile agents populated with human spatial semantics as well as means for inferring that semantics from sensory information. More specifically, an extensible approach to place classification is introduced and used for mobile robot localization as well as categorization and extraction of spatial semantic concepts from general place appearance and geometry. The models can be incrementally adapted to the dynamic changes in the environment and employ efficient ways for cue integration, sensor fusion and confidence estimation. In addition, a system and representational approach to semantic mapping is presented. The system incorporates and integrates semantic knowledge from multiple sources such as the geometry and general appearance of places, presence of objects, topology of the environment as well as human input. A conceptual map is designed and used for modeling and reasoning about spatial concepts and their relations to spatial entities and their semantic properties. Finally, the semantic mapping algorithm is built into an integrated robotic system and shown to substantially enhance the performance of the robot on the complex task of active object search. The presented evaluations show the effectiveness of the system and its underlying components and demonstrate applicability to real-world problems in realistic human settings.

Relation to WP The thesis relates mostly with Tasks 3.1, 3.3 and 3.4. It contains work prior to CogX but a large part of it was been done during CogX and it provides a lot more detail on the conceptual map than [26] does.

The thesis can be downloaded from `http://www.pronobis.pro/phd`.

References

[1] Alper Aydemir, Moritz Göbelbecker, Andrzej Pronobis, Kristoffer Sjöö, and Patric Jensfelt. Plan-based object search and exploration using semantic spatial knowledge in the real world. In *to appear in Proc. of the European Conference on Mobile Robotics (ECMR'11)*, Örebro, Sweden, September 2011.

[2] Alper Aydemir, Kristoffer Sjöö, John Folkesson, and Patric Jensfelt. Search in the real world: Active visual object search based on spatial relations. In *Proc. of the IEEE International Conference on Robotics and Automation (ICRA'11)*, 2011.

[3] Emma Brunskill, Thomas Kollar, and Nicholas Roy. Topological mapping using spectral clustering and classification. In *Proceedings of the 2007 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'07)*, pages 3491–3496, San Diego, CA, USA, October 2007.

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

[5] I. Cos-Aguilera, L. Canamero, and G. Hayes. Using a sofm to learn object affordances. In *Proceedings of the 5th Workshop of Physical Agents (WAF04)*, 2004.

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

[7] Zhan Cui David A. Randell and Anthony G. Cohn. A spatial logic based on regions and connection. In *3rd Int. Conf. on Knowledge Representation and Reasoning*, 1992.

[8] Keith S. Donnellan. Reference and definite descriptions. *Philosophical Review*, 75(3):281–304, 1966.

[9] Li Fei-Fei and Pietro Perona. A Bayesian hierarchical model for learning natural scene categories. In *Proceedings of the 2005 IEEE Conference on Computer Vision and Pattern Recognition (CVPR'05)*, 2005.

[10] Thomas David Garvey. Perceptual strategies for purposive vision. Technical Report 117, Artificial Intelligence SRI International, 333 Ravenswood Ave., Menlo Park, CA 94025, September 1976.

[11] J. Gibson. *The Ecological Approach to Visual Perception.* Lawrence Erlbaum Associates, Hillsdale, NJ., 1979.

[12] J. Kelleher. *A Perceptually Based Computational Framework for the Interpretation of Spatial Language*. PhD thesis, Dublin City University, 2003.

[13] David M. Kortenkamp and Terry 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.

[14] Steffen L. Lauritzen and Thomas S. Richardson. Chain graph models and their causal interpretations. *Journal Of The Royal Statistical Society Series B*, 64(3):321–348, 2002.

[15] 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.

[16] G.D. Logan and D.D. Sadler. *A Computational Analysis*, chapter 13. The MIT Press, 1999.

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

[18] Michael J. Milford and Gordon F. Wyeth. Mapping a suburb with a single camera using a biologically inspired SLAM system. *IEEE Transactions on Robotics*, 24(5):1038–1053, October 2008.

[19] Oscar Martinez Mozos, Cyrill Stachniss, and Wolfram Burgard. Supervised learning of places from range data using AdaBoost. In *Proceedings of the 2005 IEEE International Conference on Robotics and Automation (ICRA'05)*, Barcelona, Spain, 2005.

[20] Oscar Martinez Mozos, Rudolph Triebel, Patric Jensfelt, Axel Rottmann, and Wolfram Burgard. Supervised semantic labeling of places using information extracted from sensor data. *Robotics and Autonomous Systems (RAS)*, 55(5):391–402, 2007.

[21] J. Mugan and B. Kuipers. Continuous-domain reinforcement learning using a learned qualitative state representation. In *International Workshop on Qualitative Reasoning (QR-08)*, 2008.

[22] D. A. Norman. *The design of everyday things*. Doubleday, 1990.

[23] J. O'Keefe. *The Spatial Prepositions*, chapter 7. The MIT Press, 1999.

[24] Ivandré Paraboni, Kees van Deemter, and Judith Masthoff. Generating referring expressions: Making referents easy to identify. *Computational Linguistics*, 33(2):229–254, June 2007.

[25] Andrzej Pronobis. *Semantic Mapping with Mobile Robots*. PhD thesis, Royal Institute of Technology (KTH), Stockholm, Sweden, June 2011.

[26] Andrzej Pronobis and Patric Jensfelt. Hierarchical multi-modal place categorization. In *to appear at European Conference on Mobile Robotics*, Örebro, Sweden, September 2011.

[27] Ananth Ranganathan. PLISS: Detecting and labeling places using online change-point detection. In *Proceedings of Robotics: Science and Systems (RSS'10)*, Zaragoza, Spain, June 2010.

[28] Ananth Ranganathan and Frank Dellaert. Semantic modeling of places using objects. In *Proceedings of Robotics: Science and Systems (RSS'07)*, 2007.

[29] T. Regier. *The Human Semantic Potential*. MIT Press, 1996.

[30] Christian Siagian and Laurent Itti. Biologically-inspired robotics vision Monte-Carlo localization in the outdoor environment. In *Proceedings of the 2007 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'07)*, San Diego, CA, USA, 2007.

[31] Kristoffer Sjöö, Alper Aydemir, David Schlyter, and Patric Jensfelt. Topological spatial relations for active visual search. Technical Report TRITA-CSC-CV 2010:2 CVAP 317, KTH CSC, CAS/CVAP, SE-100 44 Stockholm, SWEDEN, September 2010.

[32] Kristoffer Sjöö and Patric Jensfelt. Learning spatial relations from functional simulation. In *to appear in Proc. of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'11)*, September 2011.

[33] Kristoffer Sjöö̸, Andrzej Pronobis, and Patric Jensfelt. Functional topological relations for qualitative spatial representation. In *Proc. of the 15th International Conference on Advanced Robotics (ICAR'11)*, Tallinn, Estonia, June 2011.

[34] D. Skočaj, G. Berginc, B. Ridge, A. Štimec, M. Jogan, O. Vanek, A. Leonardis, M. Hutter, and N. Hawes. A system for continuous learning of visual concepts. In *International Conference on Computer Vision Systems ICVS 2007*, 2007.

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

[36] Adriana Tapus and Roland Siegwart. Incremental Robot Mapping with Fingerprints of Places. In *Proceedings of the 2005 IEEE International Conference on Robotics and Automation (ICRA'05)*, Edmonton, Alberta, Canada, 2005.

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

[38] Antonio Torralba, Kevin P. Murphy, William T. Freeman, and Mark A. Rubin. Context-based vision system for place and object recognition. In *Proceedings of the 9th IEEE International Conference on Computer Vision (ICCV'03)*, 2003.

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

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

[41] Jette Viethen and Robert Dale. The use of spatial relations in referring expressions. In *Proceedings of the 5th International Natural Language Generation Conference (INLG 08)*, Salt Fork, OH, USA, June 2008.

[42] Pooja Viswanathan, Tristram Southey, James J. Little, and Alan K. Mackworth. Automated place classification using object detection. In *Proceedings of the 2010 Canadian Conference on Computer and Robot Vision (CRV'10)*, Ottawa, Ontario, Canada, June 2010.

[43] Jianxin Wu, Henrik I. Christensen, and James M. Rehg. Visual place categorization: problem, dataset, and algorithm. In *Proceedings of the 2009 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'09)*, 2009.

[44] Jianxin Wu and James M. Rehg. Where am I: place onstance and category recognition using spatial PACT. In *Proceedings of the 2008 IEEE Conference on Computer Vision and Pattern Recognition (CVPR'08)*, June 2008.

[45] Hendrik Zender and Geert-Jan M. Kruijff. Attentional anchor progression in spatially situated discourse. *Under submission*, 2011.

[46] Hendrik Zender, Geert-Jan M. Kruijff, and Ivana Kruijff-Korbayová. A situated context model for resolution and generation of referring expressions. In *Proceedings of the 12th European Workshop on Natural Language Generation (ENLG 2009)*, pages 126–129, Athens, Greece, March 2009. Association for Computational Linguistics.

[47] 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.

Search in the real world:
Active visual object search based on spatial relations

A. Aydemir, K. Sjöö, J. Folkesson, A. Pronobis, P. Jensfelt

Abstract— **Objects are integral to a robot's understanding of space. Various tasks such as semantic mapping, pick-and-carry missions or manipulation involve interaction with objects. Previous work in the field largely builds on the assumption that the object in question starts out within the ready sensory reach of the robot. In this work we aim to relax this assumption by providing the means to perform robust and large-scale active visual object search. Presenting spatial relations that describe topological relationships between objects, we then show how to use these to create potential search actions. We introduce a method for efficiently selecting search strategies given probabilities for those relations. Finally we perform experiments to verify the feasibility of our approach.**

I. INTRODUCTION

Service robots – robots that perform everyday tasks in everyday settings, whether domestic, office or other – are an eagerly anticipated goal within autonomous agent research. Compared to industrial robots, progress in service robotics has been relatively slow to date. This discrepancy is largely due to the fact that the environments service robots have to cope with are far more dynamic, unpredictable and "human-oriented" than those encountered by their industrial brethren.

Much work is going into overcoming the problem of making sense of complex environments, especially using vision. Key in this effort is the apprehension of *objects*. Objects hold an important role in human perception of space [1]. Localizing and interacting with them lies at the heart of various robotics research challenges, and while there is no shortage of open questions in dealing with objects, the bulk of previous work relies on the assumption that the particular object in question is already within the sensory reach of the robot. An often stated reason for this is tasks such as object recognition and object manipulation are already challenging enough. Nevertheless, as the field advances in its aim to build versatile service robots, the assumption of objects being readily available in the field of view of robot's sensors is no longer reasonable.

A mobile robot operating in the real world will have to interact with objects of varying size, shape and degree of mobility, to name a few complications. One way around the issue is to let the environment keep track of the objects and report their location when asked; however, this creates a dependency on an intelligent environment. Thus, it is imperative to be able to reliably locate objects visually in the real world in order to perform tasks such as place categorization, fetching and carrying, and manipulation.

The goal of object search, then, is to produce a set of sensing actions which brings the target object into the sensor's field of view. For efficiency, it should consist of a minimum number of sensing actions with maximal object detection probability. This is an example of active vision [2]. In the context of object search we refer to this as active visual search (AVS).

Considering the case of searching for a 3D object in 3D space, solving the AVS problem is far from trivial. Factors such as occlusion and illumination affect the search outcome significantly; therefore, to construct and execute such a plan, the searcher must actively adjust its sensor parameters to obtain the highest quality data. Search within the context of an agent's current sensory input has been investigated in some detail in [3], [4]. However, the problem of search on a mobile platform, in a real world environment has seen less activity.

Most significantly, the mobile AVS problem has a dimensionality proportional to the number of sensor parameters that can be actively controlled – such as the position and orientation of a sensor, for each action. Uninformed search, i.e. without any prior information on the target object's location, inevitably suffers from the curse of dimensionality. The pioneering work done by Tsotsos [5] showed that the problem of optimal search is NP-hard. [6] provided the probabilistic framework for performing object search which is based on the Bayesian theory. Using the same probabilistic framework [7] performed experiments on a humanoid robots in a real world setting. [8] presented an approach where the robot uses visual cues to further examine a particular part of the search space.

In 1976, Garvey presented *indirect search* as a way of limiting the search space [9]. Indirect search involves first locating an intermediate object in order to facilitate the search for the target object. An example of this is finding the table first and then focusing on top of the table to find a cup in a room. Later on, Wixson [10] showed that indirect search provides a significant increase in search efficiency.

More recently the AVS problem has seen increasing interest. [11] uses spatial relations to guide the search process. [12] presents a system that creates object maps. [13] applies the methodology used in a pursuit-evasion scenario to the AVS problem providing a new insight but with limited experiments. [14] studies the case where a robot simultaneously explores and searches for objects. [15] uses object co-occurrence histograms to locate objects in the environment

The authors are with the Centre for Autonomous Systems at the Royal Institute of Technology (KTH), Stockholm, SE-100 44, Sweden. This work was supported by the SSF through its Centre for Autonomous Systems (CAS), as well as by the EU FP7 project CogX and the Swedish Research Council, contract 621-2006-4520 (K. Sjöö) [aydemir,krsj,johnf,pronobis,patric]@csc.kth.se

represented as a SLAM map. [16] focuses on getting the 6DoF pose of the object and uses probability maps to plan the search.

A. Contributions

In this work we consider the case of a mobile robot looking for an object in an indoor environment. Contributions of this work are four-fold. First we provide an application of previously introduced spatial relations in a robotics framework, by basing a strategy for AVS on them. Second, we present several variants of a method for selecting a near-optimal strategy, and compare them in the context of object search. Third, we demonstrate a method for robust execution of a set of strategies, by taking into account failed strategies and re-evaluating possible strategies. Finally, we demonstrate the above ideas by implementing them on a mobile robot.

II. Search in the real world

Imagine a robot tasked with visually locating and fetching a cell phone located somewhere on a floor of a medium sized building. Without any *a priori* information on the object's whereabouts the robot, in the worst case, must cover the entire volume of each of the rooms on that floor. To accomplish this, the robot calculates a search plan which includes a series of sensing actions with the hope that this plan will lead to localizing the said object. Sensors, and in particular cameras, have a limited field of view and a particular object can only be reliably detected within some interval of ranges. Covering the entire environment will be very time consuming and appear quite inefficient to human observers. Therefore a robotic system would greatly benefit from starting its search with some initial information and thereby a non-uniform *a priori* probability distribution function (PDF) defined over the volume of search space.

Assume there is such a PDF defined over the metric space, as is done in [16], [13], [7]. A robot making use of such *a priori* information will perform better than the aforementioned uninformed search, given that its initial PDF is an accurate representation of the real world's state. However this representation of probabilities would suffer from being susceptible to small changes in the environment since no abstraction over the metric space is present. An obvious example would be an object moved from one end of a meeting table to another. Such a system would detect the absence of the object and proceed with a full-fledged search in the entire environment.

Furthermore, several different PDFs for various objects will be harder to maintain and use as the environment grows in size. This has led to experiments in the previous work being done in a very limited search space, as it is computationally expensive to run such a system in larger environments. Finally, a robot interacting with humans is likely to receive information on an abstract level and not on the metric level. Humans describe positions not by exact coordinates but by relations to other entities in the environment. Thus metric level systems will need some mechanism to help them interpret such information. On the other hand, a system that does not take into account lower level aspects might perform poorly through failing to take account of such low level factors as occlusions, limited sensor range, and illumination.

III. Problem Formulation

All of the above points out the necessity of introducing higher level abstraction to the AVS problem, while still meeting the lower-level challenges of a real world scenario. We accomplish this by introducing functional definitions of spatial relations to the AVS problem.

The main issue that this work aims to deal with is: given information – possibly uncertain – about spatial relations between objects in an environment, how is the agent to organize an efficient search aimed at finding a given object?

A. Choosing a next best view

We introduce the next best view selection algorithm following the formulation of [6]. The robot has an initial PDF over the 3D space Ψ. The search region Ψ is discretized by tessellating it into 3D cubes $c_1...c_n$. A sensing action s is then defined as taking an image of Ψ and running a recognition algorithm to determine whether the target object o is present in the image 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; $s = s(x_c, y_c, z_c, p, t, f, a)$.

An agent starts out with an initial PDF for the target object's location over Ψ. We assume that there is exactly one target object in the environment either inside or outside the search region. Let $p(c_i)$ be the probability of the object's center being in the i^{th} cell.

The next best view selection is then defined as:

$$\underset{j=1..N}{\operatorname{argmax}} \sum_{i=1}^{n} p(c_i) S(c_i, j) \tag{1}$$

Where N is the number of candidate sensing actions and S is defined as:

$$S = \begin{cases} 1, & \text{if } c_i \text{ is covered by the } j^{\text{th}} \text{ sensing action} \\ 0, & \text{otherwise} \end{cases}$$

Finally, the set of candidate view points is determined by randomly sampling the reachable space in Ψ.

B. Search Strategies and strategy steps

We wish to determine the *strategy* that minimizes the expected *cost* to find the target object. A strategy consists of a sequence of steps, each of which is a search procedure in its own right: a "simple" search, looking for an object given some specific prior probability distribution.

For example, a strategy might be composed of the steps

1) Go to room 1
2) Search for the table (which could be anywhere in the room)
3) Search for the box on top of the table
4) Search for the book inside the box

Another strategy might be

1) Go to room 1
2) Search for the box, which could be anywhere in the room but at a height compatible with being on the table
3) Search for the book inside the box

In this case, the robot uses the relational information directly, without the extra step of localizing the table.

The objective, then, is to find the most efficient sequence of steps, out of all sequences that lead to the target object.

Each strategy step, such as "the box on the table" corresponds to a 3D PDF (Figure 2). The step is carried out by greedily generating a set of potential view points, as described in III-A. The process is continued until the object is found, or the remaining probability is lower than a threshold (we set it to 30%), in which case it is deemed that the current strategy step has failed.

The cost of a strategy step is calculated as follows: First a 3D PDF corresponding to a strategy step is calculated. Then the set of next best view points that covers 70% of this PDF is generated in accordance with III-A. Finally the total distance travelled to visit all the view points in the set is calculated. The cost thus is based on the total amount of movement until a certain proportion of the initial probability is covered.

IV. SPATIAL RELATIONS AND SEARCH

Objects in environments that are created and used by human beings do not occur randomly. Rather, people design and organize spaces in ways that serve various functional purposes. This organization is expressible in terms of *spatial relations*.

Spatial relations are abstractions of the configuration in space of objects, such as their distances, directions or topological relationships. These help humans structure and remember aspects of their environment, and are likewise of great potential use when a robot has to search for objects in that same environment.

We make use of two of the most important topological spatial relations: "in" (meaning that an object is contained in the convex hull of another) and "on" (meaning that one object is being physically supported by another). The object that contains or supports is termed the "landmark", while the other is termed the "trajector".

In [11] a detailed computational model for each of these relations is proposed, and a method for computing a probability density, as might be used by an AVS procedure, is presented. These models take the form of functions that take the pose and geometry of two objects and yield a scalar measure of how applicable the relation "on" or "in", respectively, is to the configuration in question. High values mean the trajector's pose corresponds very well to being "on"/"in" the landmark. These functions, when normalized over 3D space, produce probability density functions that can be directly used in AVS.

Given the position of a landmark, this method drastically reduces the search space when looking for a trajector that has a known spatial relation to that landmark. Even when the landmark's position is not known the spatial relation information may help accelerate the search by biasing the distribution.

[11] assumes that the robot has complete knowledge of which relations hold. This is not the typical case, however; rather, what is given will be a probability distribution over possible relations, gleaned from common-sense knowledge databases or learned from experience in real environments. The question then becomes how best to investigate the different possible relational configurations in order to find the sought object at as low a cost as possible.

V. STRATEGY SELECTION

The object search strategy selection is modeled as a Markov decision process, MDP, over the belief state. The target object location is represented by an n-tuple of booleans s. Each element corresponds to a relational description of the object location such as: "book on table in livingroom". We refer to these descriptions as configurations for the object. An element of s is true if the object has the configuration in question. The configurations are not mutually exclusive. We restrict ourselves to configurations that contain a specific room. The s is a discrete random variable. Its probability evolves as the robot searches for the object. This probability is the belief state of the MDP.

The actions, a, are specific strategies for searching configurations. A single configuration will always have a direct search strategy which is to search for the object with the *a priori* distribution of object locations dependent on the configuration as a whole. Some configurations will also have indirect search strategies with several steps, as exemplified in Sec. III-B. Each action has a set of possible final states and costs. The state transitions consist of either finding the object or failing at some step. Finding the object changes the belief to certainty and ends the search; failing at some step changes the probabilities of the configurations and the costs of future actions.

The probabilities change by the Bayes update rule:

$$p(\mathrm{s}|z) = p(z|\mathrm{s})p(\mathrm{s})/p(z) \qquad (2)$$

where z is the observation of failing the current step. The probability of successfully finding the object in a step given a specific configuration is the sum of the probability mass covered by all the view cones selected during that step.

Costs of actions are based on an estimate of robot motion needed to carry out the action, including travel to the room and movement between the selected view points. After a failed step, costs for subsequent actions may be decreased, if the search located objects that are intermediate steps in those actions.

The Bellman equation without discount for this system is:

$$V(x) = \max_a (R(x,a) + \sum_{x'} p(x'|x,a)V(x')). \qquad (3)$$

Where x and x' are belief states and V is the value function which here is the negative expected cost. The maximum is taken over all actions (i.e. search strategies) and the sum is over the various possible transitions states x' from x under action a. The optimal action would be the argument of the maximum.

Without a discount on future costs and without any certainty of finding the object, a stopping criterion for the search is required, or the expected cost will be infinite. The choice of stopping criteria will affect the optimal policy. The search terminates when the probability of the object being in a room is below some threshold for every room. One could also use the probability of a configuration or just the posterior probability of the object being in the environment. A stopping criterion at the room level is useful as it allows for exploiting the separation of the search into rooms to make policy computation tractable.

We further simplify the policy calculations by limiting the state transitions to either success or failure at the final step. This means that we need not update the probabilities of the intermediate configurations, e.g., "table in livingroom".

The expected cost for an action selection policy, π is:

$$< cost|\pi >= E_0(\pi) = C_n(\pi) + Q_n(\pi)E_n(\pi) \qquad (4)$$

where $E_n(\pi)$ is the expected cost of the continued search given that the n^{th} policy action failed.

$$Q_n(\pi) = Q_{n-1}(\pi)(1 - p_n(\pi)) \qquad (5)$$

where $p_n(\pi)$ is the probability of success on the n^{th} action given that the previous actions all failed. Q_n is the probability of failing n steps. Then suppressing the dependence on the policy π:

$$C_n = C_{n-1} + Q_{n-1}(p_n c_{sn} + (1 - p_n)c_{fn} \qquad (6)$$

where c_{sn} and c_{fn} are the expected cost of success and failure respectively.

We can now compute the expected cost of any policy if we knew $E_n(\pi)$. It can be approximated based on Q_n. Q_n starts as $Q_0 = 1.0$ and is then reduced as n increases, asymptotically approaching $\bar{Q}$ which is the *a priori* probability that the object is not in the environment. Q_n is therefore a measure of the progress of our search. We will explore two assumptions on E_n: i), it is simply a constant or ii), it is proportional to $(Q_n - \bar{Q})$.

The constant assumption leads to the problem of choosing a specific constant. We use two methods of selecting this constant future cost. First, introducing a parameter $\bar{E}$ chosen based on typical search costs.

$$E_0 = C_n + Q_n\bar{E}. \qquad (7)$$

Second, use Eq.(4) to estimate the constant by setting $E_n = E_0$.

$$E_0 = C_n + Q_nE_0 \to E_0 = C_n/(1 - Q_n). \qquad (8)$$

Similarly, using the second assumption on E_n we find:

$$E_0 = C_n/(1 - Q_n(Q_n - \bar{Q})/(1 - \bar{Q})), \quad n > 0. \qquad (9)$$

Once an assumption has been made one can find a well-defined optimal action by setting the depth of the policy search, n, or the number of steps to project the MDP. The larger n is the smaller the effect of our approximation; however, the number of belief states grows exponentially with n. This makes simple exhaustive search of all paths impractical for more than a few steps. If the greedy search $n = 1$ is used the search will often be sub-optimal; for example, moving from one room to another carries a large cost and should not be chosen until the current room has been well searched, whereas the cost of returning to the room later is not considered by the greedy search.

Search over multiple rooms is complex. Nevertheless if the search were restricted to one room the sequence of actions would be independent of the state of the search in the other rooms. The magnitude of the problem may thus be greatly reduced by finding the optimal search sequence for each room separately. We then only need to optimize the choice of room at each step, knowing the sequence of actions to take given the room.

Within a room the greedy strategy works well so long as there are no dependencies between the separate configurations. Such independence does not hold in general. In some cases a configuration is made easier to search by having failed on some earlier search; e.g., finding the book on the table after having found the table on a previous failed search. These special configurations can be enumerated. We then do a restricted forward projection of the MDP within a room by choosing at each step to project the lowest expected cost policy from the previous step and the lowest cost policy out of those that contain the special configurations for each of the special configurations. In this way we are not likely to miss an advantage from these dependencies.

During policy evaluation we look at all sequences of room choices out to our search depth, choosing for each room the next action from the list of optimal actions found for that room. We then take the sequence of actions the has the lowest expected cost.

This gives us an efficient way to reduce the search over policies, breaking the problem up first into rooms and then into searches containing a limited number of policies at each step.

VI. Experiments

A. Simulation

We used simulations to verify that our action selection policies could produce lower cost searches on average. The MDP model matches the simulation while the actual implemented object search on the robot is not modeled perfectly in the MDP used for policy selection. Besides not being able to model the uncertainty in object recognition for all viewing angles, we also could not model other relatively random aspects of the real world robot, such as chance recognition of the tables and furniture that might help with later searches.

The simulations were done by varying the *a priori* distribution of the object over configurations. We set these randomly and then computed the action under a policy. We then

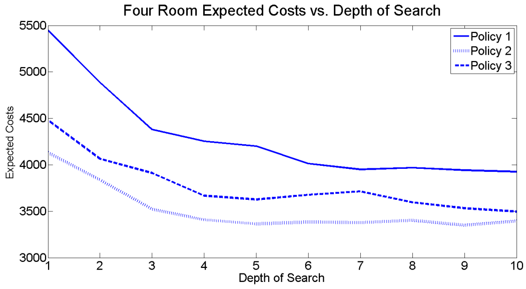


Fig. 1. We show the simulated expected costs for the example of four rooms using the three policies vs. n, the depth of the policy's search over actions.

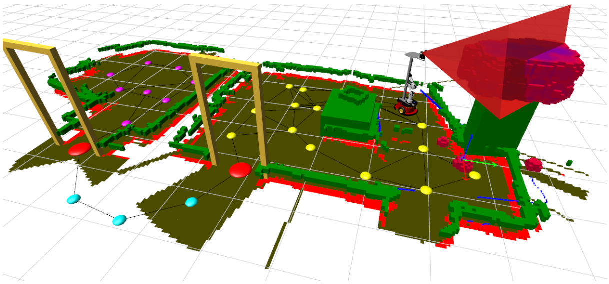


Fig. 2. An example PDF (shown in purple color) that corresponds to the relation "book on bookcase" and selected view point during an actual run. The map containing two rooms is also shown.

found the next action under the policy for the contingency of failure. We continued until the search would have ended due to our stopping criteria. Then we computed the expected cost of executing the entire sequence of actions using Eq. (4) where E_n is zero when we reach the stopping point. All policy variations were checked on this initial distribution. Finally a new random distribution was then selected and the process repeated. We performed 100 distributions and computed the resulting performance for various policies and depth of search. In this way we could see how the three assumptions on E_n, the depth of search, and the size of the environment affect the expected cost of search.

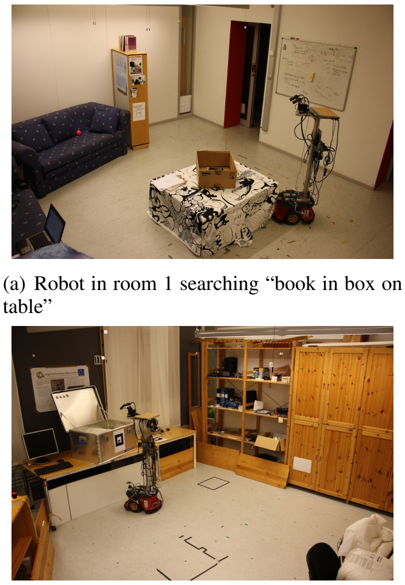
We found that the best expected costs were found by using equation (8). Figure (1) shows the expected costs vs depth of search for our policies 1, 2, and 3 using Eq. (7), Eq. (8) and Eq. (9) respectively. One can see that the cost drops about 20-25% by searching out 3 steps. It then does not change significantly by extending the search to more steps. The standard deviation at these points was about 300-450. This shows that policy 2 is expected to outperform the other two and that there is some gain to looking several steps ahead.

We also compared to exhaustive searching all actions out to three steps. This took on average 450 ms to compute an action and had an expected cost of 3900. This compares to the time 0.4 ms for our policies 1, 2, and 3 which achieved about this same level of cost at 3 steps. Exhaustive searching to two steps took 10 ms and a cost of 4250 vs. about 0.2 ms for policies 1, 2, and 3. So our exploiting the separation into rooms reduced the exponential growth from a factor of 45 to 2 even with only 4 rooms.

We also looked at much larger environments of up to 10 rooms. For instance for a 10 room environment policy 2 with 5 steps was 12% better than a 1 step policy. For 10 rooms and five steps, policies 1, 2 and 3 all took around 500 ms to compute an action at 5 step depth.

B. Robot experiments

1) Setup: In order to demonstrate its practical workability, we also implemented and tested our approach on a real-world autonomous system. Experiments were carried out on a Pioneer III wheeled robot, equipped with a Hokuyo URG laser scanner, and a camera mounted at 1.4 m above the floor. Experiments took place in two different rooms, connected by a corridor that the robot could traverse whenever it decided to search the other room. A SLAM implementation [17] carried out localization using a previously built map.



(a) Robot in room 1 searching "book in box on table"

(b) Robot in room 2 searching "book in crate"

Fig. 3. The robot used during experiments in two different rooms performing strategies

Room 1 contained the following fixed, identifiable objects: One small and one large bookcase, and one table; room 2 contained another table and a set of shelves. Mobile objects used were a cardboard box, a metal crate and a book (the target object).

The given initial belief state across configurations is presented in Table I. Note that the probabilities do not sum to 1; rather, configurations subsume each other; for example, `book ON table IN room1` contains `book IN box ON table IN room1` as a special case. (In other words, the probability that the book is on the table but *not* in the box is zero.)

2) Selected policy: The policy chosen given the same initial belief state, maps and object geometries is deterministic;

Configuration	Probability
book IN room1	0.55
book ON table1 IN room1	0.05
book ON small_bookcase IN room1	0.30
book IN small_bookcase IN room1	0.05
book IN large_bookcase IN room1	0.05
book IN box ON table1 IN room1	0.05
book IN box IN room1	0.15
book IN room2	0.40
book ON table2 IN room2	0.05
book IN crate IN room2	0.30
book IN shelves IN room2	0.05

TABLE I

INITIAL CONFIGURATION PROBABILITIES USED IN EXPERIMENTS

Go to `room1`
↓
Search for `small_bookcase`
↓
Search for `book ON small_bookcase`
↓
Search for `box`
↓
Search for `book IN box`
↓
Go to `room2`
↓
Search for `crate`
↓
Search for `book IN crate`
↓
Object not found

TABLE II

POLICY GENERATED DURING EXPERIMENTS

for the above state, the policy chosen (assuming the book was never detected) is presented in Table II. The robot performs indirect search on the most likely landmarks in the first room, then moves on to the second room. The search is aborted if it fails all three strategies, the cost-to-probability ratio for the remaining possibilities falling below the threshold.

3) Results – Accurate probabilities: The system was run 20 times, with the target object placed at each configuration a number of times commensurate with the configuration probabilities provided the robot. The results were as follows:

Configuration	Freq.	Success	Avg. views
ON sm_bookcase IN r1	6	6	2.67
IN sm_bookcase IN r1	1	1	5
ON lg_bookcase IN r1	1	0	20
IN box ON table1 IN r1	1	1	7
IN box IN r1	3	2	10
ON table2 IN r2	1	0	22
IN crate IN r2	6	5	11.17
IN shelves IN r2	1	0	19
Overall	20	15	9.6

The results show that the policy selected for the given probabilities performs well, catching most of the configurations whose cost-to-probability ratio are not below the threshold. A different threshold would naturally mean more strategies examined, and thus longer searches, but also fewer failures.

4) Results – Inaccurate probabilities: To confirm that the proposed strategy selection algorithm makes proper use of the probabilities it is given, we also carried out two tests in which the robot was provided different configuration probabilities from those listed above, producing different search policies accordingly. It was then estimated, based on this and the previous runs, how successful and costly those policies would be, given that the reality (i.e. the actual configurations) were the same as originally.

Case 1: By shifting 0.2 worth of probability mass away from "book ON small_bookcase IN room1" to "book IN large_bookcase IN room1", and similarly swapping the probabilities of "book IN crate IN room2" and "book ON table2 IN room2", a policy is generated which tries strategies in this order: First "book IN box IN room1", then "book IN large_bookcase IN room1", and finally "book ON table2 IN room2".

Case 2: Similarly, swapping "book ON small_bookcase IN room1" with "book ON table1 IN room1" and "book IN crate IN room2" with "book ON table2 IN room2" yields strategies in the sequence: "book ON table1 IN room1", "book IN box IN room1" and "book ON table2 IN room2".

When these two policies, based on modified probabilities, are applied to the set of configurations drawn from the probabilities in Table I, (simulated) success rates and view counts worsen considerably:

Run	Appr. avg. views	Appr. success rate
Accurate	9.6	75%
Inaccurate 1	19.7	25%
Inaccurate 2	12.1	20%

These results indicate that the strategy selection algorithm does indeed make proper use of the probabilistic information it is provided.

C. Conclusions

We have presented a method for robust and scalable AVS using spatial relational information. We have introduced the idea of using object-object spatial relations as an abstraction method for AVS. We showed how groupings of spatial relations can be used as search strategies in the context of AVS. Furthermore, we provide a decision theoretic strategy selection method to obtain a near-optimal search behavior and to handle cases where some strategies fail to find the target object. We have finally concluded through real world experiments the feasibility and correctness of our presented ideas.

Using spatial relational information as a way of influencing object search greatly improves both the search efficiency and outcome. However the search performs poorly when the probabilities associated with these strategies do not correspond to the real world.

D. Future Work and Discussion

Directions for future investigation involve making use of dense 3D point cloud representation of scenes to guide the search. The functional aspects of our everyday world mean that 3D structure provides better cues to object locations compared to using only visual appearance. Therefore exploiting shape properties of scenes would be beneficial for a searcher robot.

In the experiments section the qualitative object location probabilities are given to the system manually beforehand and they are not updated based on the search result. Instead of hard-coded probabilities, we would like to give the robot access to a database of spatial knowledge. Abstracting this knowledge makes the the problem tractable and the knowledge representation more robust. This knowledge can be mainly regarded as spatial common sense knowledge that is not environment specific. For instance, the category of a room can be used to build a prior over the objects that are more likely to be found in that room [18]. Such general knowledge can be learned by the robot over the course of its operation, but can also be transferred from humans either directly or by an analysis of annotated databases (e.g. LabelMe [19], ConceptNet [20]) or results gathered using Internet search engines.

It also is important to develop the methods to maintain these probabilities over very long periods of time so that the searcher robot can adapt to its environment. One future direction is designing a probabilistic graphical model representing spatial knowledge at the conceptual level in order to perform inference on object locations using common sense knowledge and various types of information acquired from the robot's sensors. We propose to structure the abstracted spatial knowledge according to a semi-probabilistic ontological representation that combines high level spatial concepts as well as relationships between those concepts and instances of objects and rooms in the environment.

In order to fully represent the statistical dependencies between the random variables expressing the uncertainties captured by the representation, we need a more expressive model such as Bayesian Networks (BN) or Markov Random Fields (MRF).

However, in order to capture the different types of dependencies that exist in the model, as a future research direction we suggest using chain graph models, being a natural generalization of the above. Chain graph models have the advantage over either BNs or MRFs of being able to express both strictly causal relationships as well as symmetric and associative relations, both of which can be identified in representation of spatial knowledge [21].

References

[1] S. Vasudevan, S. Gächter, and R. Siegwart, "Cognitive spatial representations for mobile robots perspectives from a user study," In Proc. ICRA Workshop: Semantic Information in Robotics (ICRA - SIR 2007), Rome, Italy, 2007.

[2] R. Bajcsy, "Active perception vs. passive perception," in *Proc. 3rd Workshop on Computer Vision: Representation and Control*. Washington, DC.: IEEE Press, October 1985, pp. 55–59.

[3] 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*, vol. 113, p. 2006, 2006.

[4] M. Björkman and J.-O. Eklundh, "Vision in the real world: Finding, attending and recognizing objects," *International Journal of Imaging Systems and Technology*, vol. 16, pp. 189–208, 2006.

[5] J. K. Tsotsos, "On the relative complexity of active vs. passive visual search," *International Journal of Computer Vision*, vol. 7, no. 2, pp. 127–141, 1992.

[6] Y. Ye, "Sensor planning for object search," Ph.D. dissertation, 1997.

[7] F. Saidi, O. Stasse, and K. Yokoi, "Active visual search by a humanoid robot," *Recent Progress in Robotics: Viable Robotic Service to Human*, vol. 16, pp. 171–184, 2009.

[8] 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.

[9] T. D. Garvey, "Perceptual strategies for purposive vision," AI Center, SRI International, 333 Ravenswood Ave., Menlo Park, CA 94025, Tech. Rep. 117, Sep 1976.

[10] L. E. Wixson and D. H. Ballard, "Using intermediate objects to improve the efficiency of visual search," *Int. J. Comput. Vision*, vol. 12, no. 2-3, pp. 209–230, 1994.

[11] K. Sjöö, A. Aydemir, D. Schlyter, and P. Jensfelt, "Topological spatial relations for active visual search," Centre for Autonomous Systems, KTH, Stockholm, Tech. Rep. TRITA-CSC-CV 2010:2 CVAP317, July 2010.

[12] P. Viswanathan, D. Meger, T. Southey, J. Little, and A. Mackworth, "Automated spatial-semantic modeling with applications to place labeling and informed search," may. 2009, pp. 284 –291.

[13] G. Hollinger, D. Ferguson, S. Srinivasa, and S. Singh, "Combining search and action for mobile robots," in *ICRA'09: Proceedings of the 2009 IEEE international conference on Robotics and Automation*. Piscataway, NJ, USA: IEEE Press, 2009, pp. 800–805.

[14] H. Masuzawa and J. Miura, "Observation planning for efficient environment information summarization," oct. 2009, pp. 5794 –5800.

[15] T. Kollar and N. Roy, "Utilizing object-object and object-scene context when planning to find things," in *ICRA'09: Proceedings of the 2009 IEEE international conference on Robotics and Automation*. Piscataway, NJ, USA: IEEE Press, 2009, pp. 4116–4121.

[16] J. Ma, "Real-time applications of 3d object detection and tracking," Ph.D. dissertation, California Institute of Technology, 2009.

[17] J. Folkesson, P. Jensfelt, and H. Christensen, "The m-space feature representation for SLAM," *IEEE Transactions on Robotics*, vol. 23, no. 5, pp. 1024–1035, Oct. 2007.

[18] A. Pronobis, O. M. Mozos, B. Caputo, and P. Jensfelt, "Multi-modal semantic place classification," *The International Journal of Robotics Research (IJRR), Special Issue on Robotic Vision*, vol. 29, no. 2-3, pp. 298–320, Feb. 2010.

[19] B. Russell, A. Torralba, K. Murphy, and W. Freeman, "Labelme: A database and web-based tool for image annotation," *International Journal of Computer Vision*, vol. 77, pp. 157–173, 2008.

[20] C. Havasi, R. Speer, and J. Alonso, "Conceptnet 3: a flexible, multilingual semantic network for common sense knowledge," in *Recent Advances in Natural Language Processing*, Borovets, Bulgaria, September 2007.

[21] M. Frydenberg, "The Chain Graph Markov Property," *Scandinavian Journal of Statistics*, vol. 17, no. 4, 1990.

Hierarchical Multi-Modal Place Categorization

Andrzej Pronobis[*] Patric Jensfelt[*]
[*]*Centre for Autonomous Systems, Royal Institute of Technology, Stockholm, Sweden*

Abstract— **In this paper we present an hierarchical approach to place categorization. Low level sensory data is processed into more abstract concept, named *properties* of space. The framework allows for fusing information from heterogeneous sensory modalities and a range of derivatives of their data. Place categories are defined based on the properties that decouples them from the low level sensory data. This gives for better scalability, both in terms of memory and computations. The probabilistic inference is performed in a chain graph which supports incremental learning of the room category models. Experimental results are presented where the shape, size and appearance of the rooms are used as properties along with the number of objects of certain classes and the topology of space.**

Index Terms— **place categoriation; graphical models; semantic mapping; machine learning**

I. Introduction

The topic of this paper is place categorization, denoting the problem of assigning a label (kitchen, office, corridor, etc) to each place in space. To motivate why this is useful, consider a domestic service robot. Such a robot should be able to "speak the language" of the operator/user to minimize training efforts and to be able to understand what the user is saying. That is, the robot should be able to make use of high level concepts such as rooms when communicating with a person, both to verbalize spatial knowledge but also to process received information from the human in an efficient way.

Besides robustness and speed, there are a number of additional desirable characteristics of a place categorization system:

C1: Categorization The system should support true categorization and not just recognition of room instances. That is, it should be able to classify an unknown room as "a kitchen" and not only recognize "the kitchen".

C2: Spatio-temporal integration The system should support integration over space and time as the information acquired at a single point rarely provides enough evidence for reliable categorization

C3: Multiple sources of information No single source of information will be enough in all situations and it is thus important to be able to make use of as much information as possible.

C4: Handles input at various levels of abstraction The system should not only be able to use low level sensor data but also higher level concepts such as objects.

C5: Automatically detect and add new categories The system should be able to augment the model with new categories identified from data.

C6: Scalability and complexity The system should be scalable both in terms of memory and computations. That is, for example, it should not be a problem to double the number of room categories.

C7: Automatic and dynamic segmentation of space The system should be able to segment space into areas (such as rooms) automatically and should be able to revise its decision if new evidence suggesting another segmentation is received.

C8: Support life-long incremental learning The robot system cannot be supplied with all the information at production time, it needs to learn along the way in an incremental fashion throughout its life.

C9: Measure of certainty There are very few cases where the categorization can be made without uncertainty due to imperfections in sensing but also model ambiguities. Ideally the system should produce a probability distribution over all categories, or at least say something about the certainty in the result.

In out previous work we have designed methods that meet C1, C3, C7 and partly C2, C4 and C9. In this paper we will improve on C4 and C9 and add C6 and C7. The main contribution of the paper relates to C4, C6 and C9.

A. Outline

In Section II presents related work and describes our contribution with respect to that. Section III describes our method and Section IV provides implementation details. Finally, Section V describes the experimental evaluation and Section VI draws some conclusions and discusses future work.

II. Related work

In this section we give an overview of the related work in the area of place recognition and categorization. Place categorization has been addressed both by the computer vision and the robotics community. In computer vision the problem is often referred to as scene categorization. Although also related, object categorization methods are not covered here. However, we believe that objects are key to understanding space and we will include them in our representation but will make use of standard methods for recognizing/categorizing them. Table II maps some of the methods presented below to the desired characteristics presented in the previous section.

In computer vision one of the first works to address the problem of place categorization is [19] based on the so called "gist" of a scene. One of the key insights in the paper is that the context is very important for recognition and categorization of both places and objects and that these processes are intimately connected. Place recognition is formulated in the context of localization and information about the connectivity of space is utilized in an HMM. Place categorization is also addressed using a HMM. In [23] the problem of grouping images into semantic categories is addressed. It is pointed out that many

	C1: Categorization	C2: Spatio/temporal	C3: Multi source	C4: Multi levels	C5: Novelty detection	C6: Scalability	C7: Segmentation	C8: Incremental	C9: Uncertainty
[19]	X	x							X
[23]	X								
[20]									x
[10]	X								
[12]	X	x	X	x			x		
[14]									
[9, 16]								X	
[13]									x
[26]	x		x	x					
[15]	X	x	X						x
[24]	X	x							
[18]									X
[17]	X	X					X	X	X
[22]	X								X
[21]		x			X			X	
This work	X	x	X	X		X	x	x	X

TABLE I

CHARACTERIZING SOME OF THE PLACE CATEGORIZATION WORK BASED ON THE DESIRABLE CHARACTERISTICS FROM SECTION I.

natural scenes are ambiguous and the performance of the system is often quite subjective. That is, if two people are asked to sort the images into different categories they are likely to come up with different partitions. [23] argue that *typicality* is a key measure to use in achieving meaningful categorizations. Each cue used in the categorization should be assigned a typicality measure to express the uncertainty when used in the categorization, i.e. the saliency of that cue. The system is evaluated in natural outdoor scenes. In [4] another method is presented for categorization of outdoors scenes based on representing the distribution of codewords in each scene category. In [25] a new image descriptor, PACT, is presented and shown to give superior results on the datasets used in [19, 4].

In robotics, one of the early systems for place recognition is [20] where color histograms is used to model the appearance of places in a topological map and place recognition performed as a part of the localization process. Later [10] uses laser data to extract a large number of features used to train classifiers using AdaBoost. This system shows impressive results based on laser data alone. The system is not able to identify and learn new categories: adding a new category required off-line re-training, no measure of certainty and it segmented space only implicitly by providing an estimate of the category for every point in space. In [12] this work is extended to also incorporate visual information in the form of object detections. Furthermore, this work also adds a HMM on top of the point-wise classifications to incorporate information about the connectivity of space and make use of information such as offices are typically connected to corridors. In [14] a vision only place recognition system is presented. Super Vector Machines (SVMs) are used as classifiers. The characteristics are similar to those of [10]; cannot identify and learn new categorizes on-line, only works with data from a single source and classification was done frame by frame. In [9, 16] a version of the system supporting incremental learning is presented. The other limitations remains the same. In [13] a measure of confidence is introduce as a means to better fuse different cues and also provide the consumer of the information with some information about the certainty in the end result. In [15] the works in [10, 14] are combined using an SVM on top of the laser and vision based classifiers. This allows the system to learn what cues to rely on in what room category. For example, in a corridor the laser based classifier is more reliable than vision whereas in rooms the laser does not distinguish between different room types. Segmentation of space is done based on detecting doors that are assumed to delimit the rooms. Evidence is accumulated within a room to provide a more robust and stable classification. It is also shown that the method support categorization and not only recognition. In [24] the work from [25] is extended with a new image descriptor, CENTRIS, and a focus on visual place categorization in indoor environment for robotics. A database, VPC, for benchmarking of vision based place categorization systems is also presented. A Bayesian filtering scheme is added on top of the frame based categorization to increase robustness and give smoother category estimates. In [17] the problem of place categorization is addressed in a drastically different and novel way. The problem is cast in a fully probabilistic framework which operates on sequences rather than individual images. The method uses change point detection to detect abrupt changes in the statistical properties of the data. A Rao-Blackwellized particle filter implementation is presented for the Bayesian change point detection to allow for real-time performance. All information deemed to belong to the same segment is used to estimate the category for that segment using a bag-of-words technique. In [27] a system for clustering panoramic images into convex regions of space indoors is presented. These regions correspond roughly with the human concept of rooms and are defined by the similarity between the images. In [21] panoramic images from indoor and outdoor scenes are clustered into topological regions using incremental spectral clustering. These clusters are defined by appearance and the aim is to support localization rather than human robot interaction. The clusters therefore have no obvious semantic meaning.

As mentioned above [12] makes use of object observations to perform the place categorization. In [6] objects play a key role in the creation of semantic maps. In [18] a 3D model centered around objects is presented as a way to model places and to support place recognition. In [22] a Bayesian framework for connecting objects to place categories is presented. In [26] the work in [12] is combined with detections of objects to deduce the specific category of a room in a first-order logic way.

A. Contributions

In this paper we contribute a method for hierarchical categorization of places. The method can make use of a very diverse set of input data, potentially also including spoken dialogue. We make use of classical classifiers (SVM in our case, building on the work [15]) and a graphical model to fuse information at a higher level. The categorical models for rooms are based on so called *properties* of space, rather than the low level sensor characteristics which is the case in most of the other work presented above. This also means that a new category could be defined without having the need to re-train from the sensor data level. The properties decouples the system. The introduction of properties also makes the system more scalable as the low level resources (memory for models and computations for classifiers) can be shared across room categorizers. The system we present still rely on the detection of doors like [15] but the graphical model allows us to add and remove these doors and thus change the segmentation of space. The system will automatically adjust the category estimates for each room taking into account the new topology of space.

III. Hierarchical multi-modal categorization

We pose the problem of place categorization as that of estimating the probability distribution of category labels, c_i, over places, p_j. That is, we want to estimate $p(c_i, p_j)$. We consider a discrete set of places rather than a continuous space. In our implementation the places are spread out over space like bread crumbs every one meter [26]. The places become nodes (representing free space) in a graph covering the environment. Edges are added when the robot has traveled directly between two nodes.

In our previous work [26] we performed place categorization by combining a room/corridor classifier (based on [10]) with an ontology that related objects to specific room types. For example, we inferred being in a living room if the classification system reported a room and a sofa and a TV set were found (objects associated with a living rooms according to the ontology). This method had some clear and severe shortcomings that made it only appropriate for illustrating ideas rather than being a real world categorization system in anything but simple and idealized test scenarios. Furthermore, because the system was unable to retract inferred information any categorization was crisp and set in stone. Conceptually the solution has several appealing traits. It allowed us to teach the system, at a symbolic level, to distinguish different room categories simply by assigning specific objects to them. It combined information from low level sensor data (to classify room/corridor) with high level concepts such as objects.

The place categorization system in this paper provides a principled way to maintain the advantages mentioned above even in natural environments. Our approach is based on the insight that what made the previous system easy to re-train was that the categorization was based on high level concepts rather than on low level sensor data. For this purpose, we introduce what we call *properties* of space where in the previous system the properties corresponded to the existence of certain types of objects. In general these properties could be related to, for example, the size, shape and appearance of a place.

The introduction of properties decomposes our approach hierarchically. The categories are defined based on the properties and the properties are defined based on sensor data, either directly or in further hierarchies. This is closely related to the work on part based object recognition and categorization [3]. The property based decomposition buys us **better scalability** in several ways. Instead of having to build a model from the level of sensor data for every new category, we can reuse the low level concepts. This **saves memory** (models for SVMs can be hundreds of megabytes in size) and **saves computations** (calculations shared across categories). The introduction of properties also **makes training easier**. Once we have the models for the properties, training the system for a new category is decoupled from low level sensor data. The properties can be seen as high level basis functions on which the categories are defined, providing a significant dimensionality reduction. The graph made up of the free space nodes can be used to impose topological constraints on the places as well and help lay the foundation for the segmentation process.

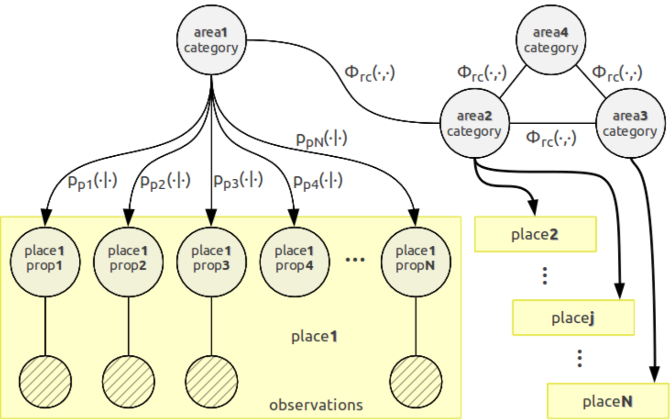


Fig. 1. Structure of the graphical model for the places showing the influence of the properties and the topology on the categorization and segmentation.

We use a graphical model to structure the problem, starting from the place graph. More precisely we will use a probabilistic chain graph model [8]. Chain graphs are a natural generalization of directed (Bayesian Networks) and undirected (Markov Random Fields) graphical models. As such, they allow for modelling both "directed" causal as well as "undirected" symmetric or associative relationships, including circular dependencies. Figure 1 shows our graphical model. The structure of model depends on the topology of the environment. Each discrete place is represented by a set of random variables connected to variables representing the semantic category of areas. Moreover, the category variables are connected by undirected links to one another according to the topology of the environment. The potential functions $\phi_{rc}(\cdot,\cdot)$ represent the knowledge about the connectivity of areas of certain semantic categories (e.g. kitchens are typically connected to corridors). The remaining variables represent properties of space. These can be connected to observations of features extracted directly from the sensory input. Finally, the functions $p_{p_1}(\cdot|\cdot)$, $p_{p_2}(\cdot|\cdot)$, $\ldots$, $p_{p_N}(\cdot|\cdot)$ model spatial properties.

The joint density f of a distribution that satisfies the Markov property associated with a chain graph can be written as [8]:

$$f(x) = \prod_{\tau \in T} f(x_\tau | x_{pa(\tau)}),$$

where $pa(\tau)$ denotes the set of parents of vertices τ. This corresponds to an outer factorization which can be viewed as a directed acyclic graph with vertices representing the multivariate random variables X_τ, for τ in T (one for each chain component). Each factor $f(x_\tau|x_{pa(\tau)})$ factorizes further into:

$$f(x_\tau|x_{pa(\tau)}) = \frac{1}{Z(x_{pa(\tau)})} \prod_{\alpha \in A(\tau)} \phi_\alpha(x_\alpha),$$

where $A(\tau)$ represents sets of vertices in the moralized undirected graph $\mathcal{G}_{\tau \cup pa(\tau)}$, such that in every set, there exist edges between every pair of vertices in the set. The factor Z normalizes $f(x_\tau|x_{pa(\tau)})$ into a proper distribution.

In order to perform inference on the chain graph, we first convert it into a factor graph representation [1]. To meet the real time constraints posed by most robotics applications we then use an approximate inference engine, namely Loopy Belief Propagation [11].

IV. Implementatation

In our implementation, each object class results in one property, encoding the expected/observed number of such objects. In addition, we use of the following properties:

- *shape* (e.g. elongated, square) –
  Extracted from laser data
- *size* (e.g. large (compared to other typical rooms)) –
  Extracted from laser data
- *appearance* (e.g. office-like appearance) –
  Extracted from visual data
- *doorway* (is this place in a doorway) –
  Extracted from laser data

In indoor environments, rooms tend to share similar functionality and semantics. In this work we cluster places into areas based on the door property of places (using door detector from [15]). The doorway property is considered to be crisp. The door places are not part of the chain graph but rather act as edges between areas. However, the graphical model allows us to easily change the topology if new information becomes available. The overall system therefore performs segmentation automatically and the dynamic nature of it is based on reevaluating the existence of doors. Figure 2 illustrates how the places (small circles) are segmented into areas (ellipses) by the existence of doors (red small circles) and how this defines the topology of the areas.

We build on the work in [15] when defining the property categorizers for shape, size and appearance (see [15] for details). The categorizers are based on Support Vector Machines (SVMs) and the models are trained on features extracted directly from the robot's sensory input. A set of simple geometrical features [10] are extracted from laser range data in order to train the shape and size models. The appearance models are build from two types of visual cues, global, Composed Receptive Field Histograms (CRFH) and local based on the SURF features discretized into visual words [2]. The two visual features are further integrated using the Generalized Discriminative Accumulation Scheme (G-DAS [15]). The models are trained from sequences of images and laser range data recorded in multiple instances of rooms belonging to different categories and under various different illumination settings (during the day and at night). By including several different room instances into training, the acquired model can generalize sufficiently to provide categorization rather than instance recognition. The estimate for the uncertainty in the categorization results is based on the distances between the classified samples and discriminative model hyperplanes (see [13] for details).

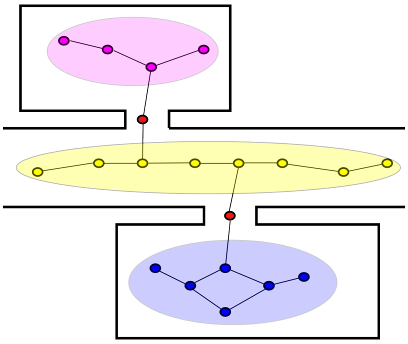


Fig. 2. The set of places, $\{p_i\}$, is segmented into areas based on the door places. The doors form the edges in the topological area graph.

To learn the probabilities associated with the relations between rooms, objects, shapes, sizes and appearances we analyzed common-sense resources available online (for details see [7]) and the annotated data in the COLD-Stockholm database[1]. The relations between rooms and objects were bootstrapped from part of the *Open Mind Indoor Common Sense* database[2]. The object-location pairs found through this process were then used to form queries on the form '*obj* in the *loc*' that were fed to an online image search engine. The number of hits returned was used as a basis for the probability estimate. Relations that where not found this way were assigned a certain low default probability not to rule them out completely.

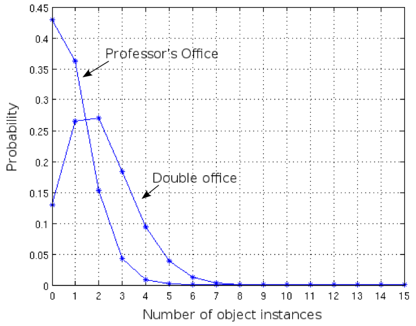


Fig. 3. The Poisson distributions modelling the existence of a certain number of objects in a room on the example of computers present in a double office and a professor's office.

The conditional probability distributions $p_{p_i}(\cdot|\cdot)$ for the object properties are represented by Poisson distributions. The parameter λ of the distribution allows to set the expected number of object occurrences. This is exemplified in Fig. 3

[1] `http://www.cas.kth.se/cold-stockholm`
[2] `http://openmind.hri-us.com/`

which shows two distributions corresponding to the relation between the number of computers in a double office and a professor's office. In the specific case of the double office, we set the expected number of computers to two. In all remaining cases the parameter λ is estimated by matching $p_\lambda(n = 0)$ with the probability of there being no objects of a certain category according to the common sense knowledge databases.

V. Experiments

A. *Experimental Setup*

The COLD-Stockholm database contains data from four floors. We divide the database into two subsets. For training and validation, we used the data acquired on floors 4, 5 and 7. The data acquired on floor 6 is used for evaluation of the performance of the property classifiers and for the real-world experiment.

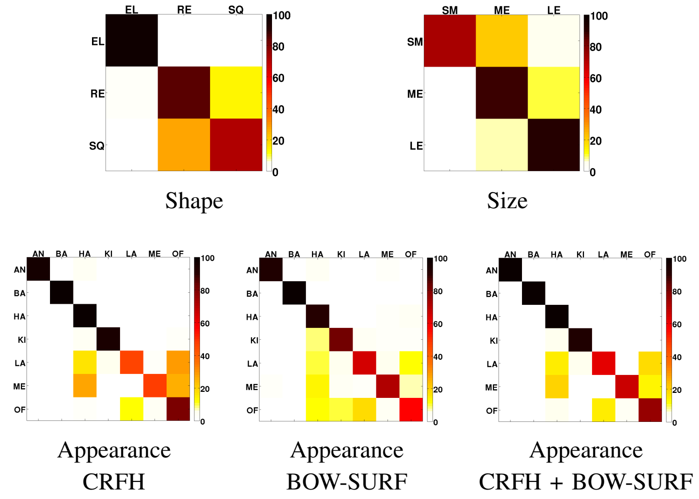
For the purpose of the experiments presented in this paper, we have extended the annotation of the COLD-Stockholm database to include 3 room shapes (elongated, square and rectangular), 3 room sizes (small, medium and large) as well as 7 general appearances (anteroom-, bathroom-, hallway-, kitchen-, lab-, meetingroom- and office-like). The room size and shape, were decided based on the length ratio and maximum length of edges of a rectangle fitted to the room outline. These properties together with 6 object types defined 11 room categories used in our experiments, see Figure 5.

B. *Evaluation of Property Categorizers*

The performance of each of the property categorizers was evaluated in separation. Training and validation datasets were formed by grouping rooms having the same values of properties. Parameters of the models were obtained by cross-validation. All training and validation data were collected together and used for training the final models which were evaluated on test data acquired in previously unseen rooms. Table II presents the results of the evaluation. The classification rates were obtained separately for each of the classes and then averaged in order to exclude the influence of unbalanced testing set. As can be seen all classifiers provided a recognition rates above 80%. Furthermore, integrating the two visual cues (CRFH and BOW-SURF) increased the classification rate of the appearance property by almost 5%. From the confusion matrices in Fig. 4 we see that the cases with confusion occurs between property values being semantically close.

Property	Cues	Classification rate
Shape	Geometric features	84.9%
Size	Geometric features	84.5%
Appearance	CRFH	80.5%
Appearance	BOW-SURF	79.9%
Appearance	CRFH + BOW-SURF	84.9%

TABLE II
Classification rates for each of the properties and cues.



Shape

Size

Appearance
CRFH

Appearance
BOW-SURF

Appearance
CRFH + BOW-SURF

Fig. 4. Confusion matrices for the evaluation of the property categorizers.

C. *Real-world experiments*

In the real-world experiment the robot was joysticked around manually. The robot started with only the models obtained in the evaluation of the property categorizers. Laser based SLAM [5] was performed while moving and new places were added every meter traveled into unexplored space. The robot was driven through 15 different rooms while performing real-time place categorization without relying on any previous observations of this particular part of the environment. The object observations where provided by human input. The information comes into the change graph in exactly the same was as would real object detections.

Figure 5 illustrates the performance of the system during part of a run. The 11 categories can be found along the vertical axis. The ground truth for the room category is marked with a box with thick dashed lines. The Maximum a posteriori (MAP) estimate for the room category is indicated with white dots. The system correctly identified the first two rooms as a hallway and a single office using only shape, size and general appearance (no objects were found). The next room was properly classified as a double office. The MAP estimate switches to professors office for a short while when one computer is found and switches back again when a second if found. After some initial uncertainty where the MAP switches category several times the next room is classified as a double office until the robot finds a computer at which point it switches to professor's office. Later the robot enters a robot lab which according to its models is very similar to a computerlab. Initially there is a slightly higher probability for the hypothesis that it is a computerlab, but once the robot detects a robot arm the robotlab hypothesis completely dominates. The next non-hallway room is a single person office currently occupied by a bunch of Master's students. Because of its current appearance, the best match is a double office. The robot continues and the rest of the categorizations are correct. The system is able to perform the categorization in real-time as can be seen these preliminary results indicate that the accuracy is quite good.

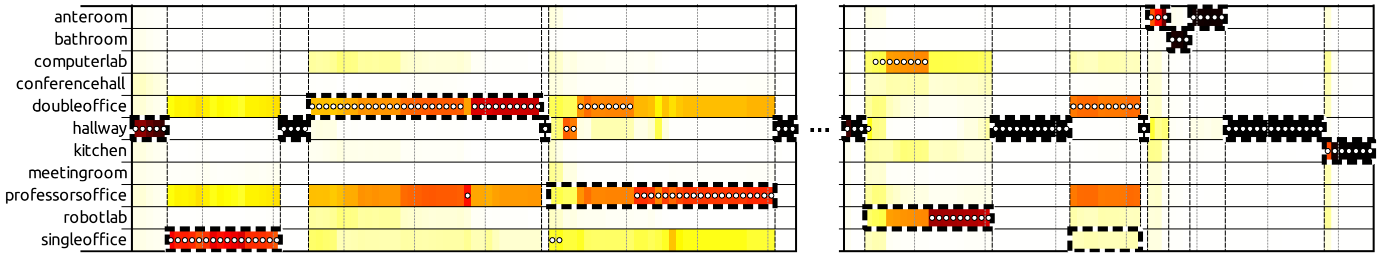


Fig. 5. Visualization of the beliefs about the categories of the rooms. The room category ground truth is marked with thick dashed lines while the MAP value is indicated with white dots.

VI. Conclusions and Future Work

In this paper we have presented a probabilistic framework combining multi-modal and uncertain information in a hierarchical fashion. So called properties were introduces as a way to model high level characteristics of the environment. These properties gave us a way to decouple the categorization into categorization of the properties based on low level sensor information and categorization of high level concepts such as rooms based on the properties. A chain graph model was used for the probabilistic inference. We provided an initial evaluation of the system which indicates that it works in well practice.

Part of the future work is to evaluate the system more thoroughly. It is important to note that we are not able to evaluate our system on other databases such as VPC [24] as it does not contain laser data. We will also investigate the use of the place categorization system in semantic mapping.

Acknowledgment

This work was supported by the SSF through its Centre for Autonomous Systems (CAS), and by the EU FP7 project CogX.

References

[1] An introduction to factor graphs. *IEEE Signal Processing Magazine*, 21:28–41, January 2004.

[2] Herbert Bay, Tinne Tuytelaars, and Luc Van Gool. SURF: Speeded up robust features. In *Proc. of ECCV'06*, 2006.

[3] G. Bouchard and B. Triggs. Hierarchical part-based visual object categorization. 2005.

[4] L. Fei-Fei and P. Perona. A bayesian hierarchical model for learning natural scene categories. In *In IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2005.

[5] J. Folkesson, P. Jensfelt, and H. I. Christensen. The m-space feature representation for SLAM. *IEEE Trans. Robotics*, 23(5):1024–1035, October 2007.

[6] C. Galindo, A. Saffiotti, S. Coradeschi, P. Buschka, J. A. Fernandez-Madrigal, and J. Gonzalez. Multi-hierarchical semantic maps for mobile robotics. In *IROS*, August 2005.

[7] Marc Hanheide, Nick Hawes, Charles Gretton, Alper Aydemir, Hendrik Zender, Andrzej Pronobis, Jeremy Wyatt, and Moritz Göbelbecker. Exploiting probabilistic knowledge under uncertain sensing for efficient robot behaviour. In *IJCAI'11*, 2011.

[8] S. L. Lauritzen and T. S. Richardson. Chain graph models and their causal interpretations. *J. Roy. Statistical Society, Series B*, 64(3):321–348, 2002.

[9] J. Luo, A. Pronobis, B. Caputo, and P. Jensfelt. Incremental learning for place recognition in dynamic environments. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS07)*, San Diego, CA, USA, October 2007.

[10] O. Martínez Mozos, C. Stachniss, and W. Burgard. Supervised learning of places from range data using adaboost. In *ICRA'05*, 2005.

[11] J. M. Mooij. libDAI: A free and open source C++ library for discrete approximate inference in graphical models. *J. Mach. Learn. Res.*, 11:2169–2173, August 2010.

[12] Oscar Martínez Mozos, Rudolph Triebel, Patric Jensfelt, Axel Rottmann, and Wolfram Burgard. Supervised semantic labeling of places using information extracted from laser and vision sensor data. *Robotics and Autonomous Systems Journal*, 55(5):391–402, May 2007.

[13] 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 (IROS07)*, San Diego, CA, USA, October 2007.

[14] A. Pronobis, B. Caputo, P. Jensfelt, and H.I. Christensen. A discriminative approach to robust visual place recognition. In *IROS'06*, 2006.

[15] A. Pronobis, O. Martinez Mozos, B. Caputo, and P. Jensfelt. Multi-modal semantic place classification. *IJRR*, 29(2-3), February 2010.

[16] 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.

[17] Ananth Ranganathan. Pliss: Detecting and labeling places using online change-point detection. In *RSS*, 2010.

[18] Ananth Ranganathan and Frank Dellaert. Semantic modeling of places using objects. In *RSS*, 2007.

[19] 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)*, pages 273–280, 2003.

[20] 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.

[21] C. Valgren and A. Lilienthal. Incremental spectral clustering and seasons: Appearance-based localization in outdoor environments. In *ICRA 2008*, pages 1856–1861. IEEE, 2008.

[22] S. Vasudevan and R. Siegwart. Bayesian space conceptualization and place classification for semantic maps in mobile robotics. *Robot. Auton. Syst.*, 56:522–537, June 2008.

[23] J. Vogel and B. Schiele. A semantic typicality measure for natural scene categorization. *Pattern Recognition*, pages 195–203, 2004.

[24] Jianxin Wu, Henrik I. Christensen, and James M. Rehg. Visual place categorization: Problem, dataset, and algorithm. In *In IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS'09)*, 2009.

[25] Jianxin Wu and James M. Rehg. Where am i: Place instance and category recognition using spatial pact. In *In IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, Anchorage, Alaska, June 2008.

[26] 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.

[27] Zoran Zivkovic, Olaf Booij, and Ben Kröse. From images to rooms. *Robotics and Autonomous Systems, special issue From Sensors to Human Spatial Concepts*, 55(5):411–418, May 2007.

Functional topological relations for qualitative spatial representation

Kristoffer Sjöö, Andrzej Pronobis and Patric Jensfelt

Abstract— **In this paper, a framework is proposed for representing knowledge about 3-D space in terms of the functional *support* and *containment* relationships, corresponding approximately to the prepositions "on" and "in". A perceptual model is presented which allows for appraising these qualitative relations given the geometries of objects; also, an axiomatic system for reasoning with the relations is put forward.**

We implement the system on a mobile robot and show how it can use uncertain visual input to infer a coherent qualitative evaluation of a scene, in terms of these functional relations

I. Introduction

Having already made great inroads into industrial settings, robotics is now making an effort to enter into environments such as homes, offices or hospitals. These kinds of spaces are, more than anything, *human-oriented,* that is constructed by and for people, used and modified by people, and occupied by people.

As a result, nearly every aspect of those spaces is shaped by the propensities, preferences and mental habits of human beings. From this association, they take on human *semantics* [1], [2], semantics that must be internalized by any robot that is to have a chance of interacting meaningfully with such environments and their occupants.

An important part of this semantics is *spatial relations*. Spatial relations are abstract, functional relationships between entities in space; they show themselves in the way humans speak about space [3], [4], albeit in a limited fashion. Inspired by these psycholinguistic clues, this work aims to imbue a robot with the ability to understand space in terms of two of the most important spatial relations in the human repertoire – "on" and "in". It proposes computational models as well as a first-order logic axiomatic system for the spatial abstractions that underlie these ubiquitous expressions. We demonstrate by experiment that the approach is suitable for automatic extraction of scene descriptions from undertain visual perception.

A. Functional relations

We humans speak of, and think of, reality in certain terms because those terms are useful to us. Abstractions permit us to make sense of the endless variability of the world, allowing for structured learning, planning and communication. Spatial relations are no exception. They represent some aspect of the environment that has functional relevance – if there was none, they would not be used and thus not learned [3], [5]. Related is the notion of object *persistence*, meaning that objects are expected to remain in the same qualitative relation over time, even if the exact geometrical positions change [6].

Functions may be things such as transporting ("groceries in a bag"), protecting ("trophies in a display case"), allowing to dry ("clothes on a line"), communicating a location ("the door on your right") – or any number of others. The variation among these functionalities is infinite; nevertheless, studies of different languages [7], [8] have indicated that there are recurring patterns, clusters of abstract functionality that are instantiated and extended in different ways for different languages and situations.

This work centers on two such clusters: mechanical support and containment, corresponding – although not perfectly – to the English prepositions "on" and "in" respectively. The importance of these concepts, evident from language, as well as their topological nature provide a hint to their potential for organising the world in a manner that can be shared between robots and people.

B. Related work

There has been research into quantifying spatial relations previously. [9] uses results from brain research to isolate geometrical factors that are important to some relations. [10] introduces a computational model in the *Attention Vector Sum*, verifying it against actual human responses. Another model is suggested by [11] in the form of spatial templates, prototypes which can provide a more or less accurate match to a situation. [12] and [13] both present graphical systems in which spatial relations are used for interaction with a user.

None of the above investigate the functionally important topological spatial relations nor are their approaches based on a functional conceptualization, something we believe important as explained above.

Topological relations are surveyed in [14]. One well-known approach is *Region connection calculus* and its variants, which provide a language for expressing qualitative relationships between regions – such as containment, tangential contact etc. Although there is some overlap with the qualitative axioms introduced below, RCC is purely geometrical and does deal with functional relationships.

This paper builds upon initial work published in [15], [16]. These earlier efforts concentrate on only one of the topological relations (ON), whereas the present work introduces the IN relation and proposes an axiomatic system detailing the relationship between IN and ON, providing the means for qualitative high-level reasoning to incorporate topological information.

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 and the Swedish Research Council, contract 621-2006-4520 (K. Sjöö)

C. *Organisation of this paper*

In Section II the concept of topological spatial relations is explained and the specific instances ON and IN are introduced. Section III details a set of first-order logic axioms structuring the relations, and Section IV shows how those axioms can be included in a probabilistic reasoning framework. Section V describes the system as implemented on a mobile robot and verifies its function experimentally. Finally, conclusions and ideas for future work are contained in Section VI.

II. TOPOLOGICAL SPATIAL RELATIONS

Spatial relations represent the configuration of a focus object, or *trajector*, relative to one or more other objects termed *landmarks*. In language, spatial relations are typically divided into different groups based on the salient geometric relationship: *Projective* spatial relations constrain the trajector's location within an essentially *directed* region relative to the landmark. The direction may depend on many factors, such as intrinsic properties of either object, or the frame of reference of an onlooker. 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 and the location of an observer. Typical examples are "on", "at" and "inside". Topological relations seem to be among the first to be learned in humans [17]. Topology is very useful for structuring space in a systematic, hierarchical way, allowing us to put together sentences such as "my keys are in a briefcase on the desk in my office on the second floor at our branch in New York". This hierarchical property makes for efficient storage and inference. For this reason, this paper focuses on the arguably most significant topological relations, "on" and "in".

A. ON

1) Ideal schema: The word "on" in English carries a central functional meaning: *support* against gravity. This encompasses for example "the book on the table", "the fly on the wall", "the ring on the finger". Other languages extend the concept differently [18], but the support criterion remains central.

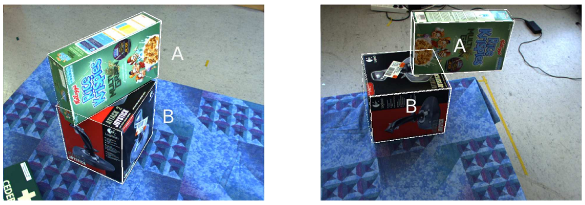
Support goes together with other functional aspects, such as *location control*. Location control imposed by one object on another means that the latter moves together with the former, such as is the case with e.g. trays, plates, buses and trains. Other connotations such as attachment or "weighing down" also overlap with the central "support" notion.

2) Computational model: Although mechanical support provides an objective criterion for defining a spatial relation, it is not typically possible for a robot to ascertain that one object is in fact supporting another. Even humans use perceptual models to estimate this, and those models may sometimes fail (see Fig. 1). We have previously suggested a computational model for a robot to be able to make such an estimate from vision [16] – briefly, three numerical criteria are weighed together to produce a quantitative function ON estimating how well one object supports another:

- *Distance*: Since the objects must touch in order for one to support the other, apparent separation between objects (as well as apparent interpenetration) penalizes the function.
- *Stability*: As the (apparent) center of mass of an object moves beyond the area of contact with its support (as in Fig. 1(b)), the function value is decreased.
- *Verticality*: When the contact surface between objects is horizontally oriented, the function is high, dropping off as the surface becomes more vertical.

Using this computational model, it is possible both to evaluate a perceived configuration of objects in terms of how well they correspond to the support relation, and to estimate the most likely configuration if the support relation is given.

This model is restricted to cases when an object is being supported "on top of" another, as opposed to hanging or adhesive support; the latter entail entirely different geometries and would need a separate perceptual model.



(a) Obvious support relation

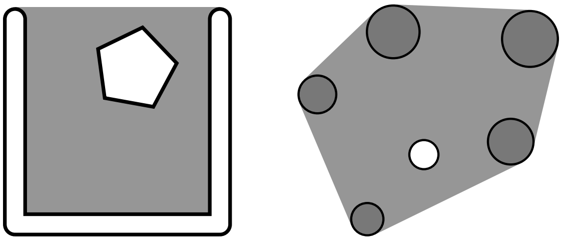
(b) Incongruent-seeming support relation

Fig. 1. Estimation of support through vision is imperfect

B. IN

1) Ideal schema: "In" as a word has a wider variety of connotations than "on" does. Besides location control and object persistence, "in" often entails aspects of concealment, protection, constraint among others. This variety of meanings is difficult to pin down precisely, but a robust approximation can be found in the idea of *containment*.

Containment signifies the inclusion of most or all of an object within the interior of another object or group of objects. "Interior" is not itself unambiguous, but even with a simple interpretation such as the convex hull, many if not most situations represented by "in" can be covered; Fig. 2 shows two examples of this.

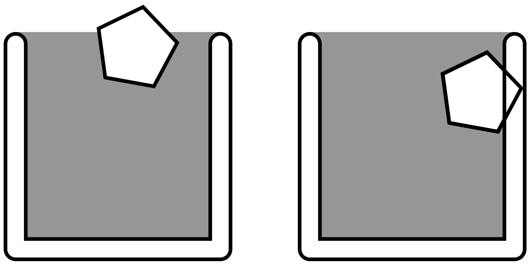


(a) Shape "in" box

(b) Circle "in" group

Fig. 2. Convex hull defining "in"

2) Computational model: Containment is computed directly as the proportion of an object O that falls within the convex hull of the container object C (see Fig. 3(a)). This proportion is termed $\textsc{In}_{con} \in [0, 1]$.



(a) Shape partly "in" box

(b) Shape and box interpenetrating

Fig. 3. Penalties on "in" estimate

However, if this were the only factor determining degree of containment, cases where O and C overlap in space – which is not physically plausible (Figure 3(b)) – would be evaluated the same as realistic configurations. Because such cases are bad examples of the relation, the model is supplemented with a penalty function on apparent object interpenetration:

$$\textsc{In}_{pen} \triangleq \begin{cases} 1 & d \geq 0 \\ e^{d/k} & d < 0 \end{cases} \tag{1}$$

where d is the minimum distance between O and C (as defined in Sec. II-A.2) and k a falloff constant.

The total estimate function for the containment spatial relation is taken to be:

$$\textsc{In} \triangleq \textsc{In}_{con} \cdot \textsc{In}_{pen} \tag{2}$$

Both "on" and "in" carry a plethora of additional, metaphorical and indirect meanings that transfer some of the concrete aspects mentioned above into other domains than space by analogy: "on my side", "in theory". Although these are illustrative of the thought processes that support spatial relations and interesting in their own right, the present work shall restrict itself to concrete, spatial usage.

III. Axiomatic System

One of the main uses for a model that can translate a geometrical relationship between perceived objects into qualitative spatial relations (and back) is to perform high-level reasoning. In order to permit that, a set of rules, or axioms, for the relational predicates is required.

Here follows a suggestion for such an axiomatic system, involving the predicates $\text{On}(x, y)$ and $\text{In}(x, y)$, which are first-order symbols corresponding to the support and containment relations described in Section II. As is inevitable with abstract reasoning, the axioms represent an idealization that will not always apply to the real world. They are reasonable approximations, however, and may be included selectively depending on the application.

Support tends to be *transitive*: if z supports y and y supports x, then z supports x as well. This is obviously not covered by the computational model in Sec. II-A; therefore, a third relation symbol is introduced, termed On_t (for "transitive On"), the properties of which are derived from the axioms.

A. Basic axioms

$$\begin{aligned} \text{On}_t(x,y) &\rightarrow \neg\text{On}_t(y,x) \quad (3)\\ \text{In}(x,y) &\rightarrow \neg\text{In}(y,x) \quad (4)\end{aligned}$$

- (3): Support is antisymmetric
- (4): Containment is antisymmetric

The above also entail irreflexivity ($\neg\text{On}_t(x,x)$, $\neg\text{In}(x,x)$)

B. Transitivity axioms

$$\begin{aligned} \text{On}(x,y) &\rightarrow \text{On}_t(x,y) \quad (5)\\ \text{On}_t(x,y) \wedge \text{On}_t(y,z) &\rightarrow \text{On}_t(x,z) \quad (6)\\ \text{In}(x,y) \wedge \text{In}(y,z) &\rightarrow \text{In}(x,z) \quad (7)\end{aligned}$$

- (5): Direct support implies transitive support.
- (6): Support is transitive – if y takes the weight of x, and z the weight of y, then that will include x as well.
- (7): Containment is transitive; this is a reasonable assumption given simple geometry and the definition of $\textsc{On}$.

C. Interchangeability axioms

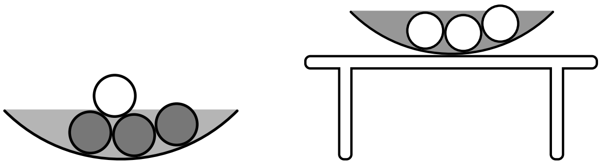
$$\begin{aligned} \text{On}_t(x,y) \wedge \text{In}(y,z) &\rightarrow \text{In}(x,z) \quad (8)\\ \text{In}(x,y) \wedge \text{On}_t(y,z) &\rightarrow \text{On}_t(x,z) \quad (9)\end{aligned}$$

$$\begin{aligned} \text{On}_t(x,y) \rightarrow\ & \text{On}(x,y) \\ \vee\ & \exists z.\, ((\text{On}(x,z) \wedge \text{On}_t(z,y)) \quad (10)\\ \vee\ & (\text{In}(x,z) \wedge \text{On}_t(z,y))) \\ & \exists y.\, (\text{On}_t(x,y) \vee \text{In}(x,y)) \quad (11)\end{aligned}$$

- (8): "Generous containment". Typically containment will physically prevent objects from sticking out. This means supported objects will also be contained.
  One consequence of this axiom is that geometrical containment may be violated for In in some cases. Figure 4(a) illustrates, however, that even in such cases functional aspects such as location control, confinement and so forth are often largely preserved and so we tend to extend the use of the word "in" to these cases as well. The axiom is thus intuitively justifiable.
- (9): "Containment provides support". When an object is contained by another, as a rule it is prevented from contact with outside objects and so must receive its supporting force directly or indirectly from the container, as illustrated in Figure 4(b).
- (10): "Support requirement". This is the necessary condition that corresponds to the sufficient conditions in Eqs. (5), (6) and (9), and asserts that an object must

be supported directly by *some* object in order to be indirectly supported.

- (11): "Base requirement". Every object must be supported by some other object.



(a) The ball is "in" the bowl (b) The balls are "on" the table

Fig. 4. Effect of interchangeability axioms

D. Hierarchy axioms

$$\mathrm{On}(x,y) \wedge (y \neq z) \quad \rightarrow \quad \neg\mathrm{On}(x,z) \qquad (12)$$

$$\mathrm{On}_t(x,y) \wedge \mathrm{On}_t(x,z) \quad \rightarrow \quad \mathrm{On}_t(y,z) \vee \mathrm{On}_t(z,y) \qquad (13)$$

$$\mathrm{In}(x,y) \wedge \mathrm{In}(x,z) \quad \rightarrow \quad \mathrm{In}(y,z) \vee \mathrm{In}(z,y) \qquad (14)$$

The hierarchy axioms ensure that the spatial relations form a tree-like structure, which is useful for representation and reasoning.

- (12): Asserts uniqueness of (direct) support. The intuitive justification for this assumption is that an object often is substantially supported by only one other object, and the *majority* of its support nearly always comes from one source.
- (13): Extends the unique-support assumption to the indirect support On_t.
- (14): Although situations can be constructred wherein two containers overlap such that each contains an object, while neither contains the other, such situations are uncommon in practice. Factors such as location control are also unlikely to be present in such cases[1].

E. Using the relational axioms

The axioms proposed in the preceding sections are valuable when processing spatial knowledge on a qualitative level.

A few examples:

- Transitivity and interchangeability axioms allow for deducing In and On_t relations even where not directly given by the computational models.
- Incomplete and qualitative knowledge can be used to guide active search for an object; for example, learning from different sources that "the bowl is on the table" and that "the apple is in the bowl", the robot can search for the table in order to help find the apple.
- Concrete-support and hierarchy constraints provide the possibility of learning about spatial relations through the process of elimination, given a closed-world assumption.
- Hierarchy constraints furthermore ensure that relations form a tree-like structure and thus make for compact storage (only a few relations need be stored whereas the rest can be deduced), as well as the potential for effectivizing algorithms operating on this knowledge.

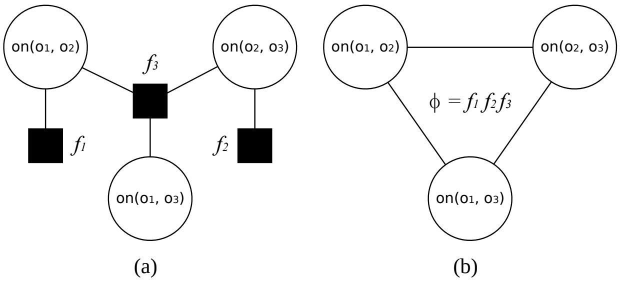


Fig. 5. Factor graph representing "on" object relations connected with a transitivity axiom (a) and a corresponding undirected graphical model (b).

In a practical application, obviously a great deal of instance knowledge will apply in addition to the axioms. Many pairs of objects will be patently impossible in the context of On and In; a room cannot be "in" a desk, and that desk can probably not be "on" an apple. Such commmonsense knowledge can be added to the knowledge base to reduce the space of possibilities. Also, for practical applications some objects (such as the floor or the room) must be exempt from Eqns. 10 and 11, as an infinite number of objects would be required otherwise.

IV. Probabilistic inference

In real-world scenarios, the information about objects perceived by a robot is inherently uncertain. This makes it important to provide the ability to transform the axioms defining object relations into a form that permits probabilistic reasoning and integration with probabilistic models such as directed or undirected probabilistic graphs [19]. Here, we introduce a probabilistic representation of axioms and show that such representation can be automatically generated according to the uncertain perception of a scene.

A. Factor-based Representation of Axioms

There is no straightforward way of defining a probabilistic interpretation of the axiomatic system presented above. Except for the fact that configurations contradicting the axioms perforce must have probability 0, nothing is said about the relative likelihoods of permitted configurations. Expressing the axioms through conditional probabilities as in e.g. a Bayes Net [19] will be non-trivial and potentially inefficient, since the relationships expressed are not causal in nature and introduce a great deal of circular cross-dependencies.

One way of introducing probabilities is to use *factor graphs* [20]. Factor graphs are bipartite graphical models, where random variables are represented using variable nodes, connected to each other not directly but via *factor nodes* –

[1] Eqn. (14) implies that, in Fig. 4(a), $\mathrm{On}_t(Ball, Bowl)$ must hold. While this rings true as regards mechanical support, one would not likely say that "The ball is on the bowl". Here, "in" takes linguistic precedence. However; while this paper gets inspiration from language, it is not primarily about modeling language *per se*.

see Fig. 5(a). Each factor node j defines a function f_j on its connected variables X_j; the joint probability is expressed as

$$p(x_1, \ldots, x_n) = \prod_j f_j(X_j)$$

This factorization makes it easy to encode the various constraints provided by the axioms. For example, if $\mathcal{O}$ is the set of objects, Eqn. (5) becomes

$$\forall \langle o_1, o_2 \rangle \in \{\mathcal{O} \times \mathcal{O}\} : \quad f_{(5)} = \begin{cases} 0, & \begin{array}{r} \text{On}_t(o_1, o_2) \\ \wedge\neg\text{On}(o_1, o_2) \end{array} \\ 1, & \text{otherwise} \end{cases}$$

Similarly, each axiom can be modeled as a factor on every applicable tuple with a value of 0 or 1. The tuples may prove intractable in some cases, such as Eqn. (10). Here, it may be necessary to reduce the set of tuples by heuristically excluding combinations that are impossible, depending on the domain. A typical example would be to divide objects into a group of base objects (e.g. a table) and mobile objects (e.g. a book) and exclude the cases when a base object is On or In any of the mobile objects.

Apart from these axiomatic factors, “probabilistic” factors can be introduced on relations and tuples of relations for which probability needs to be modeled.

An example:

$$\forall \langle o_1, o_2 \rangle \in \{\mathcal{O} \times \mathcal{O}\} : \quad f^\star = \begin{cases} \alpha_1, & \begin{array}{rl} & \text{In}(o_1, o_2) \\ \wedge & \textsc{Book}(o_1) \\ \wedge & \textsc{Library}(o_2) \end{array} \\ \alpha_2, & \ldots \end{cases} \tag{15}$$

The above encodes the likelihood, all other things being equal, that objects of different categories are inside containers of different categories. Note that the α:s are not probabilities *per se*; rather they are parameters that, in conjunction with other factors, influence the probabilities of their associated tuples in a systematic way. These parameters are prime candidates for learning. They might also be influenced by other sources of knowledge such as commonsense knowledge about typical man-made environments.

B. Automatic Generation of the Factor-based Representation

We have shown that it is possible to establish a direct correspondence between object relations and factor graph variables as well as relation axioms and factor graph factors. This can be used to design an automatic procedure converting an uncertain perception of a visual scene into a probabilistic model performing scene understanding. In the sequel, we propose such a procedure.

Our method takes as input the set of objects, enumerates all object pairs and posits a relation for each pair and relation type. In order to make the reasoning more efficient, it is possible to additionally exclude certain relations which are *a priori* impossible, such as On(A,A). The algorithm subsequently incorporates observations of given object relations, obtained by analysing the visual input as outlined in Section II. Those observations are provided in the form of values in the range $[0, 1]$ quantifying each of the perceived relations. The data flow through the system is presented in Fig. 6.

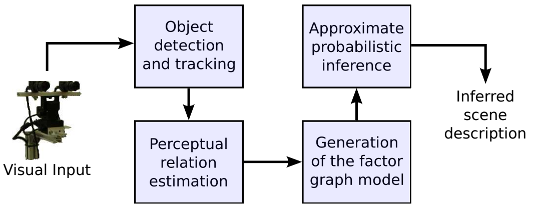

Object detection and tracking
Approximate probabilistic inference
Inferred scene description
Visual Input
Perceptual relation estimation
Generation of the factor graph model

Fig. 6. Data flow through the scene description estimation system.

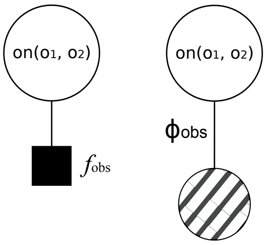


Fig. 7. An excerpt from an undirected graphical model and a corresponding factor graph illustrating the way the uncertain observations of object relations are included

The algorithm iterates over the possible relations and generates factor graph variables accordingly. Then, it analyses all relation sets matching any of the axioms specified in Section III and introduces an axiom factor for each of them. Finally, factors representing observations are generated for those relations for which the observations are available, as presented in Fig. 7. The following section show that the resulting representation may be successfully applied to the problem of understanding real-world scenes in the presence of uncertain perception.

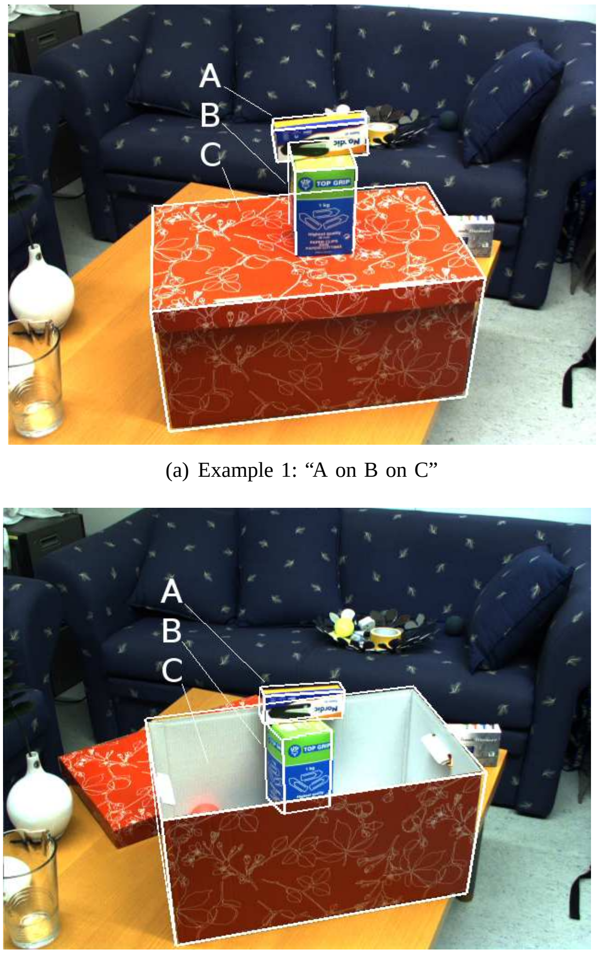
V. Experiments

In order to show how the system proposed in the preceding sections could be used in robotics applications we have implemented it on a mobile robot. The platform used is a Pioneer III wheeled robot, equipped with a a camera mounted at 1.4 m above the floor. In this experiment the robot was controlled manually so as to place the objects within the view of the camera. We assume the geometries of the objects are known in advance, but not their positions nor the qualitative relations between them.

A. Vision

For detection and pose estimation of objects, we are using a system developed at Vienna University [21]. In it, objects are detected using SIFT features trained from a variety of view points; this also provides an initial pose estimate. The pose is refined and tracked using edges and textures.

Given the estimated poses and the known geometries of the detected objects, the perceived values for the functions On and In were computed as described in Section II. Because of noise in the pose estimates, the values obtained fell



(a) Example 1: “A on B on C”

(b) Example 2: “A on B in C”

Fig. 8. Examples of consistent scene evaluation

	Example 1		Example 2	
	Per	Inf	Per	Inf
On(A,B)	92.5%	TRUE	98.9%	TRUE
On_t(A,B)		TRUE[1]		TRUE[1]
In(A,B)	0%	FALSE	0%	FALSE
On(A,C)	4.4%	FALSE	95.2%	FALSE[4]
On_t(A,C)		TRUE[2]		FALSE
In(A,C)	0%	FALSE	16.2%	TRUE[3]
On(B,A)	0%	FALSE	2.1%	FALSE
On_t(B,A)		FALSE		FALSE
In(B,A)	0%	FALSE	0%	FALSE
On(B,C)	96.4%	TRUE	1.7%	FALSE
On_t(B,C)		TRUE[1]		FALSE
In(B,C)	0%	FALSE	99.9%	TRUE

TABLE I

EXAMPLE 1, 2 EVALUATION. “PER” STANDS FOR PERCEIVED VALUE, “INF” FOR INFERRED TRUTH VALUE.

[1] Using Eqn. 5. [2] Using Eqn. 6. [3] Using Eqn. 8. [4] Using Eqn. 13.

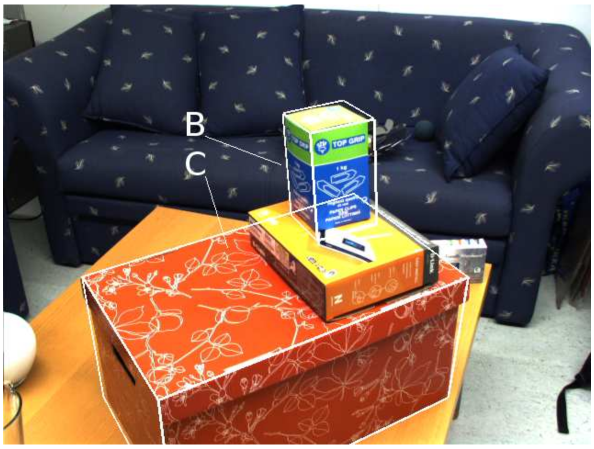


Fig. 9. Example 3: an ambiguous scene

within the continuous range $[0, 1]$. Figure 8 shows examples of scenes and Table I the extracted relation values.

B. Inference

Using the set of detected objects and their perceived relation values, the scene was instantiated as a factor graph (Section IV). Each possible relation pair In(x, y), On(x, y), On$_t(x, y)$ was instantiated as a node in the graph, as were the axioms – except that the box “C” was considered a “base object”, exempting it from appearing as the first argument in any relation and from needing a support.

The observed values of ON and IN were included as well, as unary factors working on the corresponding nodes. Inference was then performed and the maximum *a posteriori* (MAP) estimate obtained.

C. Results

Figure 8 shows two examples of scenes for which visual processing and inference were performed. The wireframe boxes indicate the object tracker’s estimated pose of each object. Table I shows the perceived as well as the inferred values for the relations.

Note that the resulting maximum-a-posteriori solutions obey the axioms. In Example 1, it can be seen that On_t is deduced in accordance with Eqs. 5, 6. Example 2 shows the effect of the interchangeability and hierarchy axioms. Here, both On(A,B) and On(A,C) are indicated by vision, but Eqn. 13 forbids them to be true simultaneously, unless On_t(B,C). Since A already has a support, B, On(A,C) is inferred to be false rather than setting On(B,C) to true. Note also that In(A,C) is made true by Eqn. 8.

In Example 3 (Figure 9), failure to recognize an object means that the object B is seemingly without a proper support. Nevertheless, Eqn. 11 causes On(B,C) to be inferred as the only consistent explanation.

It is seen that the proposed method does indeed produce

	Example 3	
	Per	Inf
On(B,C)	36.9%	TRUE
On_t(B,C)		TRUE
In(B,C)	0%	FALSE

TABLE II

EXAMPLE 3 EVALUATION

consistent qualitative descriptions of a scene, even in the presence of uncertainty, helping to bridge the gap between sensors and metric representations on the one hand and high-level reasoning on the other.

VI. Conclusions and Future Work

In this work we have suggested the use of functional, topological relations based on the notions of support and containment in order to structure spatial knowledge for autonomous robots. An axiomatic system was suggested, consisting of rules that model the first-order logical properties of the abstract relations and that will aid in high-level cognitive activities concerning space. We have demonstrated an implementation of the theory on a real robot and shown that it yields consistent results.

Spatial relations have already been put to use in object search [16], [22], wherein the relations were assumed to be given in advance. A natural next step is to use the axioms to infer the relations likely to hold in a scene and thus create priors for unseen objects, or to aid in tracking.

Another avenue of inquiry is integrating the concepts with computational linguistics, which is appropriate since the this work draws inspiration from language. Spatial relations are important for giving instructions or asking questions about objects; this work should help a robot determine which questions to ask and how to incorporate the answers into its knowledge.

The use of factor graphs to represent the relations and axioms permits their integration with more complete and expressive models directly indicating the type of the modeled relationships between random variables and clearly representing conditional independence between them. As future work, we intend to integrate the factor graph representation of axioms with a complete conceptual spatial knowledge representation within a single chain graph model [23]. Chain graphs are probabilistic graphical models providing a generalization of directed (Bayesian Networks) and undirected (Markov Random Fields) graphical models. As such, chain graphs allow for modeling both “directed” causal as well as “undirected” symmetric or associative relationships, including circular dependencies. In the context of the chain graphs, the presented representation becomes a powerful tool for reasoning about object relations that can easily be incorporated into a more complete probabilistic environment models such as the one presented in [24].

Obviously, this paper has only scratched the surface of the rich repertoire of spatial relations that humans use. Though the schemata of support and containment are doubtless very important, many others as important remain unmodeled out there. It is our belief that the function-based treatment given the relations in this paper can successfully be applied to them as well, helping to build understanding of the world that surrounds us and our robots.

References

[1] Y.-F. Tuan, *Space and Place*. University of Minnesota Press, 1977.

[2] T. Jordan, M. Raubal, G. B., and M. Egenhofer, “An affordance-based model of place in gis,” in *8th Int. Symposium on Spatial Data Handling (SDH’98)*, 1998.

[3] S. Levinson, *Space in Language and Cognition: Explorations in cognitive diversity*. Cambridge University Press, 2003.

[4] K. Coventry and S. Garrod, *Saying, seeing and acting : the psychological semantics of spatial prepositions*. Hove, 2003.

[5] C. Vandeloise, *Spatial Prepositions: a case study from French*. The University of Chicago press, 1991.

[6] D. B. M. Haun, J. Call, G. Janzen, and S. C. Levinson, “Evolutionary psychology of spatial representations in the hominidae,” *Biology*, 2006.

[7] S. Levinson and S. Meira, “natural concepts in the spatial topological domain – adpositional meanings in crosslinguistic perspective: An exercise in semantic typology,” *Language*, vol. 79, no. 3, 2003.

[8] A. Herskovits, *Language and Spatial Cognition*. Cambridge University Press, 1986.

[9] J. O’Keefe, *The Spatial Prepositions*, ch. 7. The MIT Press, 1999.

[10] T. Regier and L. A. Carlson, “Grounding spatial language in perception: An empirical and computational investigation,” *Journal of Experimental Psychology*, vol. 130, no. 2, pp. 273–2098, 2001.

[11] G. Logan and D. Sadler, *A Computational Analysis*, ch. 13. The MIT Press, 1999.

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

[13] J. Kelleher, *A Perceptually Based Computational Framework for the Interpretation of Spatial Language*. PhD thesis, Dublin City University, 2003.

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

[15] K. Sjöö, A. Aydemir, T. Mörwald, K. Zhou, and P. Jensfelt, “Mechanical support as a spatial abstraction for mobile robots,” in *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*, October 2010.

[16] A. Aydemir, K. Sjöö, and P. Jensfelt, “Object search on a mobile robot using relational spatial information,” in *Proceedings of the 11th International Conference on Intelligent Autonomous Systems (IAS’10)*, (Ottawa, Canada), September 2010.

[17] J. Piaget and B. Inhelder, *The child’s conception of space*. Routledge, 1948.

[18] S. Levinson, *Encyclopedia of cognitive science*, ch. Spatial Language. Nature Publishing Group, 2003.

[19] C. M. Bishop, *Pattern Recognition and Machine Learning*. Springer, 2006.

[20] F. R. Ksischang, B. J. Frey, and H.-A. Loeliger, “Factor graphs and the sum-product algorithm,” *IEEE TRANSACTIONS ON INFORMATION THEORY*, vol. 47, pp. 498–519, February 2001.

[21] T. Mörwald, J. Prankl, A. Richtsfeld, M. Zillich, and M. Vincze, “BLORT - The Blocks World Robotic Vision Toolbox,” in *Best Practice in 3D Perception and Modeling for Mobile Manipulation (in conjunction with ICRA 2010)*, 2010.

[22] K. Sjöö, A. Aydemir, D. Schlyter, and P. Jensfelt, “Topological spatial relations for active visual search,” Tech. Rep. TRITA-CSC-CV 2010:2 CVAP317, Centre for Autonomous Systems, KTH, Stockholm, July 2010. http://www.csc.kth.se/˜patric/publications/cvap317-avs.pdf.

[23] S. L. Lauritzen and T. S. Richardson, “Chain graph models and their causal interpretations,” *Journal Of The Royal Statistical Society Series B*, vol. 64, no. 3, pp. 321–348, 2002.

[24] M. Hanheide, N. Hawes, C. Gretton, A. Aydemir, H. Zender, A. Pronobis, J. Wyatt, and M. Gbelbecker, “Exploiting probabilistic knowledge under uncertain sensing for efficient robot behaviour,” in *Proceedings of the 22nd International Joint Conference on Artificial Intelligence (IJCAI’11)*, (Barcelona, Spain), 2011.

Learning spatial relations from functional simulation

Kristoffer Sjöö and Patric Jensfelt
Centre for Autonomous Systems,
Royal Institute of Technology
SE-100 44 Stockholm, Sweden
[krsj,patric]@csc.kth.se

***Abstract*— Robots acting in complex environments need not only be aware of objects, but also of the relationships objects have with each other. This paper suggests a conceptualization of these relationships in terms of task-relevant functional distinctions, such as support, location control, protection and confinement. Being able to discern such relations in a scene will be important for robots in practical tasks; accordingly, it is demonstrated how predictive models can be trained using data from physics simulations. The resulting models are shown to be both highly predictive and intuitively reasonable.**

I. INTRODUCTION

As robots begin to make their way into homes, workplaces and public spaces in order to aid us in an ever-widening variety of tasks, these environments will pose a mounting challenge due to their complexity, unpredictability and scale. Unlike typical industrial and laboratory settings, environments where humans live and act are not structured in a way that is amenable to hard-coded behaviors, nor do they allow many of the simplifying assumptions that aid robots in research and manufacturing settings.

Fortunately, this does not mean that robots must act within total chaos. Human-inhabited spaces are in fact structured, but not according to any strict metrical system; instead, patterns of *functional* relationships dominate. We construct buildings and design furniture and utensils with specific purposes in mind. We interact with objects and places in order to carry out tasks great or small in scope, and therefore tend to organise the space around us to support those interactions.

Consider, for example, a kitchen. Kitchens are set up so as to support, above all, the task of preparing food. Within them, different compartments – cabinets, drawers – exist to aid in finding the right items quickly, as well as to protect them from light or air or children. Packages protect and contain foodstuffs that would spill out or mix together without them; work surfaces allow for manipulating items in a stable fashion at the right height; trays and platters help transport many objects at a time.

All of the above are examples of functional spatial relationships. Because such relationships are key to humans' interaction with an environment, robots must understand them in order to be of use to humans in that environment. Possessing a conceptual "toolbox" of functional spatial relations will allow an agent to:

1) *Isolate the relevant aspects of a process it observes*
   E.g.: Observing a human demonstrating a household task, the robot will be able to divide it into abstract, generalizable steps – "A goes *on top* of B, then B goes *through* C"
2) *Apply abstract task knowledge to novel objects and situations*
   E.g.: A robot will know that placing items on top of something, such as a tray, allows them to be transported simultaneously.
3) *Store and process spatial knowledge more efficiently*
   E.g.: Over a long period of operation in a household, instead of tracking the exact metric location of objects, a robot might represent only qualitative transitions such as an object being put inside a cabinet or on a shelf. This also helps in learning rules for how the world works and generalizing knowledge to new situations.
4) *Transmit and receive qualitative spatial knowledge in communication with humans*
   E.g.: A robot may be required to describe the location of an item to a human, or carry out a task from a verbal or written instruction; in these cases it will be crucial for it to understand spatial language, such as prepositions ("on", "through") et cetera.

However, it is not sufficient to imbue an agent with a "theoretical" concept of a functional relation. Understanding a concept implies the ability to make use of it in practice, and to do this an agent must be able to link the abstract concept with the real world, through perception and action. The purpose of this work is to demonstrate how such an understanding can be acquired by a robot, by learning to predict the qualitative outcome of basic actions as a function of the perceivable geometric relationship between objects, in terms of their shapes and positions. We are trying to find the common denominators, out of a high-dimensional space of features extracted from the scene, that contribute to this prediction. The features thus isolated, and the ability to use them to predict outcomes, together represent an understanding of a particular aspect of space.

For example, one such model would estimate, given the visual perception of the objects, whether moving one object will cause another to move as well. Being able to make this distinction will be crucial for interacting with scenes and for understanding what a human demonstrator is doing in a scene.

Constructing spatial understanding out of experience ideally entails forming concepts from scratch, from sensory-motor association and up. Carrying out such a wholesale attempt on a real robot would involve random exploration, perceptual clustering, then generating actions from scratch – as well as the practical issues of robot experiments – which represents a daunting undertaking. This work nevertheless suggests that such an effort is in fact feasible; working in simulation we show that it is possible to learn models of functional spatial relations from experience.

To accomplish this, we generate random scenes in a physics simulator (see Fig. 1), perform "micro-experiments" on the objects in those scenes and observe the outcomes. Using this information, a training process both determines the relevant features and obtains model parameters that make it possible to predict the outcome of the micro-experiment; in other words, a model for a functional relation is learned.

A. *Related work*

Attempts to quantify spatial relations for use in robotics have been made in the past. In previous work (Sjöö et al. [13]) we propose a functionally conceived model for the word "on", a topological relation. Similarly, Regier and Carlson [12] use a quantitative model for projective relations such as "above". In both of the above the models are given, not learned. Kelleher [8] provides a good survey of computational models for spatial relations. On the whole, the models described in the literature are based on geometrical, not functional aspects.

The learning of spatial relations has been tackled in e.g. Regier [11] and Skočaj et al. [14], who learn to recognize spatial relations and associate them with words, though the features used are simplistic. Examples of using robot-centered and task-related criteria in learning can be found in Ek et al. [5] and Uğur et al. [16], but these do not deal with relations.

The approach in this paper has a lot to do with the idea of *affordances* introduced by Gibson [7]. An affordance is the property of an object allowing an agent to perform a specific action or task with that object. In Gibson's concept, affordances are independent of the knowledge or predisposition of an agent; in contrast, Norman [10] instead espouses a view where an agent must be aware of the capabilities of an object for an affordance to exist. Affordances learned from experience necessarily follow this latter view.

Affordance learning has been attempted by e.g. Cos-Aguilera et al. [3], where an agent learns to recognize objects that afford "eating", "shelter" and "interaction" in a simulated environment. Mugan and Kuipers [9] learn an abstracted model that supports formation of qualitative rules for actions in a simulated "baby-chair" robot setting. The contribution of this work is the application of those same principles to spatial relations: the acquisition of functional understanding from experience, in a bottom-up fashion; obtaining specific models for several relations in a simulated setting.

B. *Outline*

The rest of this paper is structured as follows: In Sec. II we discuss what we mean by functional spatial relations, and explain the choices of candidate relations for learning, based on previous work and physical intuitions. Sec. III details the simulation framework used and the functional relationships that we use as test bed for learning. The experimental setup, including the learning procedure, is described in Sec. IV and the results of our experiments are given in Sec. V. Sec. VI, finally, summarizes the work and discusses its application as well as future avenues of research.

II. Spatial Relations

This work attempts to learn models for functional spatial relations from experience. "Functional" here should be understood in the sense of an affordance. Cos-Aguilera et al. [3] puts it in these terms:

> "the relationship between the regularities in the sensory flow of an agent and the action potentials these offer to that particular agent"

However, here we deal not with an affordance of a specific object but rather of the configuration of objects relative each other. The affordance applies with respect to some action that the agent might attempt with those objects. The configuration might hinder the action, such as when an object that the agent is trying to grasp is blocked by another; or, it might help it, as when the agent moves a number of cups by moving the tray they are standing on.

Here, we are limiting ourselves to pairwise relations. We are also assuming that the following are the only inputs given:

- Action
- Geometry of the objects
- Pose of the objects

These are obviously drastic simplifications. There are important mechanical properties such as mass, friction and elasticity which are also important for action outcomes, yet largely impossible to determine visually. Moreover, an instantaneous snapshot of a scene hides powerful dynamical cues that a human would use in order to reassess actions, such as objects wobbling or sliding. Nevertheless, humans readly make judgements regarding spatial relations even for static scenes; thus this simplification is not unreasonable.

In linguistics, "spatial relation" refers to the words and constructions used in human language to communicate important aspects of objects' locations and configurations in space. Examples from English are "to the left of", "above", "forward", "in". Although this paper does not deal directly with language, concentrating instead on relationships grounded in (simulated) experience, the language humans use in talking about space provides important clues to the sorts of functions that are relevant to people and thus worth investigating.

We have previously proposed hard-coded models for the relations "on" and "in" for use in knowledge representation (Wyatt et al. [18]) and visual search (Aydemir et al.[2]). The choice of these particular relations stems from their commonness and their strong functional connotations; see Coventry and Garrod [4]. In functional terms, "On" represents *attachment* and *support*; "in" *protection*, *concealment*, *constraint*

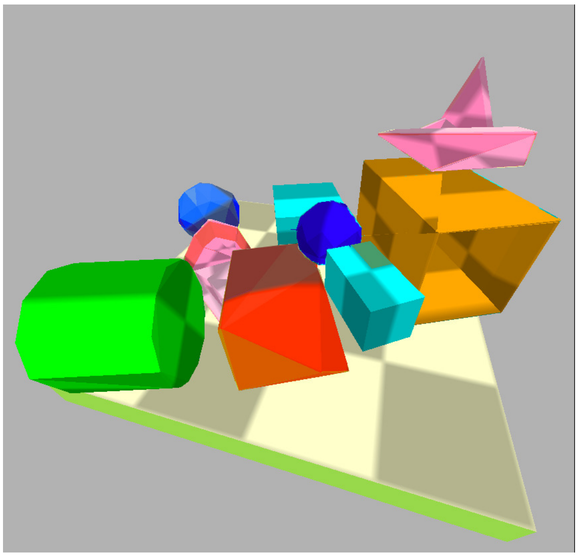


Fig. 1. Example of scene with randomized objects

(Vandeloise [17]). Both also carry a strong connotation of *location control*; i.e. the tendency for one object to cause another to move as it moves.

These aspects of the relationships between objects are of obvious use to agents as they interact with the sort of objects and facilities that abound in our everyday surroundings. Accordingly, we have chosen from among them the concepts for which to attempt to learn models in this paper.

III. FUNCTIONAL SIMULATION

As explained above, although performing experiments on a real robot is the best way to obtain data on real-world functional spatial relations, such experiments would entail a host of practical problems of their own. Simulations on the other hand allow full control over experimental conditions and are ideal for testing new methods. Therefore we have chosen simulated experiments for the purposes of this paper.

A. Simulation environment

The simulation contains a square immovable "table", above which a random set of rigid body objects are generated and allowed to fall freely onto the table and each other. Any objects that fall off the table are automatically replaced above it.

Figure 1 shows an example of a random scene. The objects generated are of the following classes:

- Solid box (cyan)
- Solid cylinder (green)
- Solid sphere (blue)
- Solid random convex polyhedron (red)
- Hollow box, one face removed (orange)
- Hollow random convex polyhedron, faces removed on one side (pink)

The training procedure follows the scheme:

1) Create random objects
2) Wait until scene becomes static
3) Record features of static scene
4) Perform "micro-experiment" and observe outcome, defining the ground truth to be learned
5) Repeat until enough data is collected
6) Train predictor

B. Functional micro-experiments

Five experiment suites are carried out on the randomly generated physical scenes. Each examines pairs of objects in the scene in terms of some task-relevant functional distinction. The distinctions are picked from the connotations of "in" and "on" suggested in Sec. II:

1) Support
2) Location control
3) Protection
4) Constraint

Attachment is disregarded because it cannot be modeled without considerably increasing the complexity of the physics simulation; moreover the function of attachment is largely overlapping with location control. Concealment is also left out as it requires explicitly simulating an agent's perception in addition to the physics. Apart from difficulties in simulating these functions there is no particular reason why the learning process should not work for them as well. "Support" is split into two separate connotations that we term "supporting force" and "causal support", explained below.

1) Supporting force: An object o' supporting another o implies that the former is inhibiting the natural tendency of the latter to fall down under the influence of gravity. Countering the force of gravity in this way means that o must, directly or indirectly, affect o' with a force that has the opposite direction from the force of gravity. The support force is relevant to task outcomes such as whether an object will damage another or prevent its movement by weighing it down.

We define the *support force* relation ground truth in the following way:

$$\textsc{For}(o,o') \triangleq \frac{\sum_{c\in C_{o,o'}} \max(0, \mathbf{f}_c \cdot -\hat{\mathbf{g}})}{\sum_{o''\neq o}\sum_{c\in C_{o,o''}} \max(0, \mathbf{f}_c \cdot -\hat{\mathbf{g}})} \tag{1}$$

where $C_{o,o'}$ is the set of contacts between objects o and o', each contact applying the force $\mathbf{f}_c$ on o, and $\hat{\mathbf{g}}$ is the direction of gravity. In other words, $\textsc{For}(o,o')$ is the proportion of all the upward-directed (i.e. supporting) forces acting on o that come from o'.

2) Causal support: By "causal support" we mean simply the fact that removing object o' causes object o to fall down or be otherwise disturbed. This relationship is important to tasks concerned with the stability of objects, including both stacking and tearing down.

It is tested through simply removing individual objects from the physics simulation and observing which of the remaining objects move more than a threshold distance as a result. The causal support relation ground truth is defined:

$$\textsc{Sup}(o,o') \triangleq \begin{cases} 1, & \text{if } o \text{ moves} \\ 0, & \text{otherwise} \end{cases} \tag{2}$$

3) Location control: As mentioned above, location control signifies the relationship where object o' moving causes object o to move along with it. This is significant when the task requires moving several objects simultaneously, or conversely when trying to separate objects from each other.

The micro-experiment used to evaluate location control consists in selecting a *mover* object o' and moving it away kinematically (irresistibly) along a straight line. The mover follows a "minimum jerk" velocity profile, modeling the way a robot or human hand might move in an actual task (see e.g. Flash and Hogan [6]).

At the end of the movement, location control is considered to hold for any object o that has moved at the same velocity as o' during most of the micro-experiment. To obtain a ground truth we perform multiple movements in random directions, resetting the scene each time, and average the result.

$$\textsc{Loc}(o,o') \triangleq \frac{1}{N} \sum_{i=1\ldots N} loc_{o,o'}(i) \tag{3}$$

where $loc_{o,o'i}(i)$ is 1 if o's velocity is equal to that of o' to within a threshold t, for at least 90% of the duration of trial i; otherwise, it is 0. For our experiments, $N = 10$.

4) Protection: Protection refers to an object o' preventing contact between object o and other external objects or agents. This distinction has task relevance when o is fragile or valuable or otherwise prone to external interference as well as, more importantly, when the agent itself needs to manipulate an object without obstruction.

In simulation we represent outside disturbance by "throwing" a small dynamic sphere in a trajectory that will hit o unless obstructed by o'. All objects in the scene except o' and o are made permeable to the sphere. The result is averaged over a series of trials to yield the ground truth:

$$\textsc{Pro}(o,o') \triangleq \frac{1}{N} \sum_{i=1\ldots N} pro_{o,o'}(i) \tag{4}$$

where $pro_{o,o'}(i)$ is 1 if the sphere contacts o' and subsequently fails to hit o. N is set to 20 in this experiment.

5) Constraint: The relationship of constraint pertains to an object o' preventing another o from moving freely. Constraint is a relevant aspect of a scene when o has motive power, or when movements or vibrations of the reference frame may cause o to roll or slide around.

We test for constraint by applying a constant force on o in a random (horizontal) direction, causing it to start moving. If it escapes the static scene, falling off the table, it is not constrained; if it fails to escape the scene is reset and the micro-experiment repeated, removing one of the objects o' that o contacted in the first iteration. If o can escape when o' is missing, o' is considered to be constraining o. Again, this is repeated several times and the result averaged.

$$\textsc{Con}(o,o') \triangleq \frac{1}{N} \sum_{i=1\ldots N} con_{o,o'}(i) \tag{5}$$

where $con_{o,o'}(i)$ is 1 if o escapes with o' absent but not with it present. N is set to 20.

C. Features

Because the intention of the proposed training procedure is to create models autonomously, we wish to bias the learning as little as possible. Accordingly, a large number of features are recorded from the static scene for each pair of objects. The features are based only on the geometry and pose of the objects, and could in principle be extracted from vision alone. They are numerical values that are more or less obvious encodings of the absolute and relative positions of the objects and of their points of contact, avoiding features that make explicit assumptions about the classes of shapes involved. Some are represented in both Cartesian, spherical and cylindrical coordinates and thus many of the degrees of freedom in the feature space encode the same information and will obviously be redundant, leaving it up to the learning process to determine which to keep and which to discard.

The full feature vector used in this paper has 93 dimensions and is composed of the following components.

($\mathbf{r}_o$ signifies the geometrical centroid of o.)

Feature	DOF #
Pose of o	1-12
Pose of o'	13-24
$\mathbf{r}_o - \mathbf{r}_{o'}$	25-32
$\lvert\mathbf{r}_o - \mathbf{r}_{o'}\rvert$	33
AVS between body of o' and $\mathbf{r}_o$	34-39
Closest separation	40
Contact normal of closest contact	41-45
Total contact patch area on o	46
Total contact patch area on o'	47
Weighted[1] average z-component of contact patch normals on o	48
Weighted[1] average z-component of contact patch normals on o'	49
AVS from contact patches on o to $\mathbf{r}_o$	50-55
AVS from contact patches on o' to $\mathbf{r}_o$	56-61
Weighted average dot product of contact patch normals	62
Vector between closest contact point and $\mathbf{r}_o$	63-71
Vector between closest contact point and $\mathbf{r}_{o'}$	72-80
Containment of o within o'	81
Vector sum from contact patches on o to $\mathbf{r}_o$	82-87
Vector sum from contact patches on o' to $\mathbf{r}_o$	88-93

[1] Weighted by contact patch area

"Vector sum" signifies a vector-valued integral over the area of each contact patch. AVS or "Attention vector sum" (Regier and Carlson [12]) is similarly a vector-valued integral but weighted with an exponential falloff that assigns more importance to space near the point of least separation between the bodies. "Containment" signifies the percentage of o's volume that falls within the convex hull of o'.

A constant term is also included to allow the model to compensate for any bias in the features.

D. Learning framework

The objective of the learning process is to produce, for each of the functional relations defined in Sec. III-B,

- A subset of the features, that is sufficient to predict whether the relation will obtain for a pair of objects in a novel scene (R_s, R_l etc.)
- A set of weights for the chosen features that gives rise to the best predictor

To this end, we train a logistic regression classifier, using the Sparse Bayesian approach – see Tipping and Faul [15]. The ground truth measures from training are thresholded to produce a binary-valued target vector $\mathbf{t}$. We consider this target vector binominally distributed given a logistic weighting together of the feature values ϕ:

$$P(\mathbf{t}|\mathbf{w}) = \prod_{n=1}^{N} \sigma(\phi_n^T \mathbf{w})^{t_n} \left[1 - \sigma(\phi_n^T \mathbf{w})\right]^{1-t_n} \qquad (6)$$

where N is the number of training examples.

The algorithm performs an iterative Bayesian update on the weights, which drives some of them to 0, leaving only those that carry significant predictive information. This makes for a sparse model. Using these weights, prediction can be performed by inserting a novel vector ϕ (containing only the selected features) into Eq. 6 and thresholding the result. Alternatively, the unthresholded value may be used as a measure of confidence.

IV. EXPERIMENTS

Simulation and learning were carried out separately for each of the 5 functional relations described in Sec. III-B. For each, 5000 random scenes were generated, consisting of 2-10 rigid bodies, which set of bodies was replaced by a new random set every 5 scenes. We use the Bullet physics engine [1] for our experiments. The friction coefficient was set randomly for each object; density was uniform and restitution (elasticity) was set to 0 to for optimal simulation quality.

In total, around 200 000 pairs of bodies were processed, each including the feature values from the static scene as well as the ground truth for the relation, as evaluated in simulation (Sec. III-B). Before features were computed, the objects' true poses were perturbed by a small Gaussian-distributed positional error, simulating sensor error for increased realism and robustness of the learned model.

K-fold cross-validation was used, with K=10. For each fold, the Sparse Bayes learning algorithm was used to learn a set of weights as explained above. 50 000 training examples were used in the training set in each fold, with the (relatively sparse) positive examples distributed equally across folds.

The algorithm sometimes failed to converge; in these cases learning was re-run with randomized initial weights.

Although the training procedure permitted real-valued targets in principle, in the following experiments the training targets were thresholded at 0.5, making the problem one of two-label classification and allowing for the drawing of Receiever operating characteristic (ROC) curves.

V. RESULTS

The ROC curves of the learned models are shown in Fig. 2. All the models perform very well on validation data, with little variance, showing the essential learnability of the relations using the features used. It is apparent that the LOC, SUP and FOR models are more distinctive; this is due to the higher degree of dichotomy in the training data, with strong clusters for both 0 and 1, whereas PRO and CON had a more indistinct distribution with fewer clearly positive examples.

In Figure 3(a), for comparison we show the results of training a model on a depleted training set, with scenes lacking one object type (general solid convex polyhedra), and evaluating it on the original data set. The curves display good performance, indicating some capacity for generalization in the models learned.

A. Feature selection

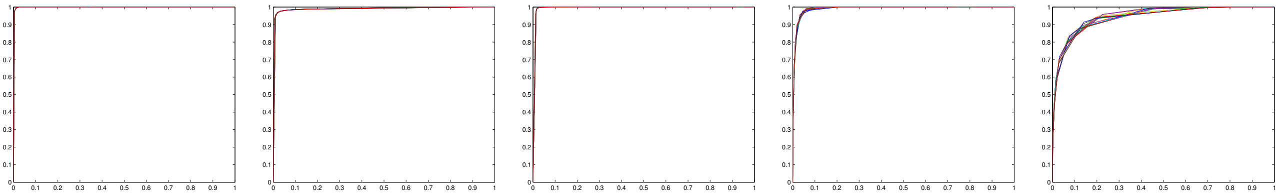
Figures 3(b) through 3(f) shows the feature weights attained for each of the relations, along with the standard deviation across the cross-validation sets. Numbers indicate indices into the feature vector; see Sec. III-C. Note the considerable stability of features used, both across cross-validation sets and between the different relations.

Some patterns can be seen, as to the features that emerge as significant (roughly, more than a standard deviation from 0); mostly they can be argued to be intuitively reasonable:

- A constant bias term (#0): tends to classify a relation negatively unless strong positive indications exist
- Mostly, the absolute positions and orientations of the bodies (#1-24) have little influence; this supports the intuition that relative, not absolute features are central.
- Separation (#40) and distance between COMs (#28): Intuitively, a functional relation is more likely to hold if objects are contacting or at least close.
- Properties of contact patches – size (#46, 47), inclination (#43, 45, 48, 49) – carry information about the physical interaction between bodies.
- Vertical alignment of bodies and contact points (#30, 31, 58, 62, 68, 69, 74, 77, 78, 84, 87) entails stability of a configuration.
- Containment (#81) is an especially important cue for protection and constraint.
- Distance from contacts to object COMs (#66, 69, 75, 78) are an indication of how much influence a contact has on the movement of an object.
- As might be expected, features encoding directions in the horizontal plane – i.e., x, y components in Cartesian coordinates or ϕ in spherical/cylindrical – carry no discriminative power because of symmetry and are generally discarded. This shows that the learning process is indeed capable of selecting those features that help it evaluate the functional properties of the scene.

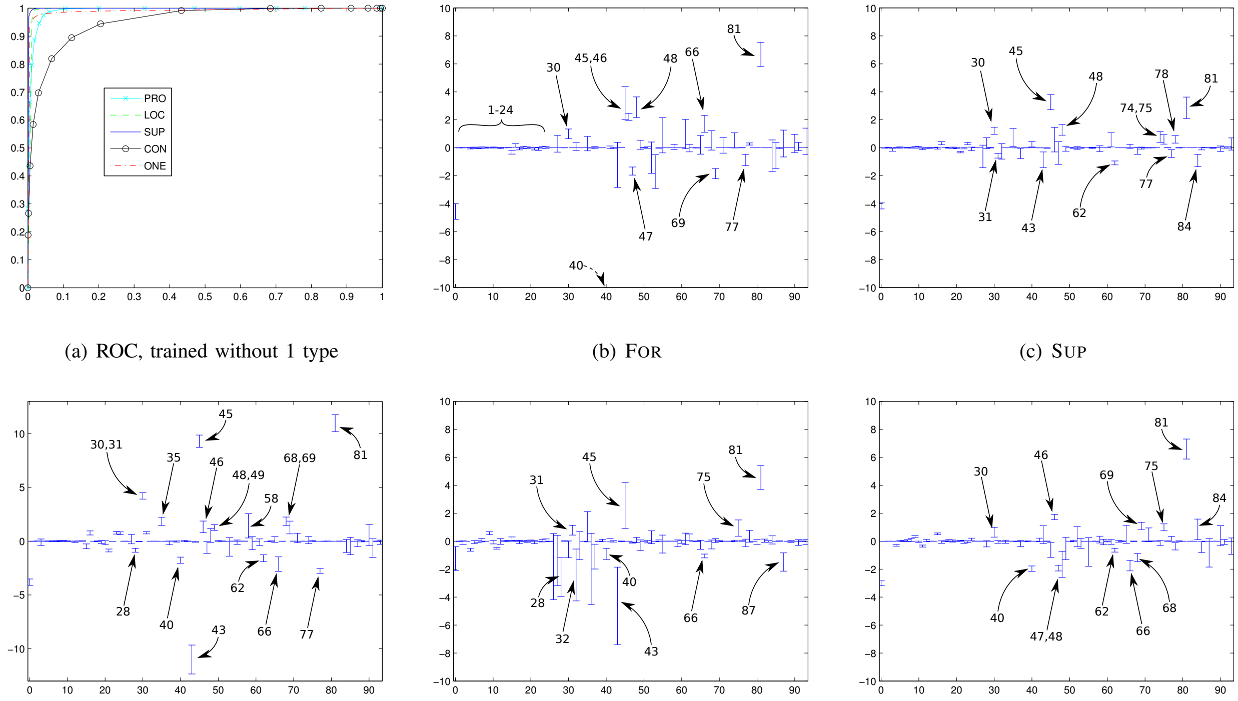
B. Example

Figure 4 shows qualitatively how the learned models classify a novel random scene. All object pairs are listed, along with the (unthresholded) outputs from the logistic function



(a) FOR (b) SUP (c) LOC (d) PRO (e) CON

Fig. 2. ROC curves for trained models. The 10 cross-validation folds are superimposed.



(a) ROC, trained without 1 type

(b) FOR

(c) SUP

(d) LOC

(e) PRO

(f) CON

Fig. 3. (a): Models trained with one object type missing, validated on scenes with all types. (b)-(f): Average weights per feature, with standard deviations.

(Eq. 6). Intuitively, these results indicate that location control and support together do in fact correspond to what we call "on" whereas "in" carries additionally the connotations of protection and constraint.

VI. CONCLUSION

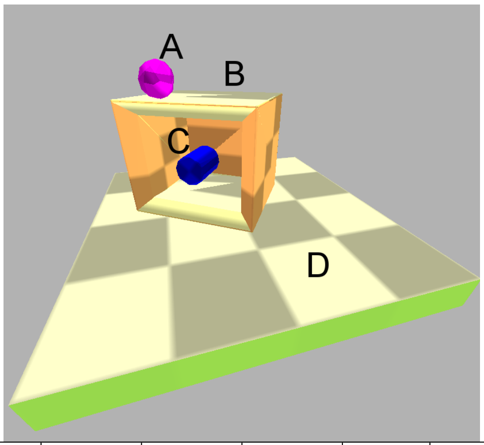
In this paper, we have put forth an approach for learning functional spatial relations from simulated experiments. Five functional distinctions were learned: effective support, support force, location control, confinement and protection. The learned predictive models are accurate within the context of the simulated environment, and shown to yield intuitively reasonable outputs on sample pairs; furthermore, they are sparse, which shows that the method is capable of extracting those features that are of functional relevance.

The unavoidable restrictions on possible object geometries and scene configurations, as well as the imperfections of the physics simulation will limit the applicability of the trained models in the real world. More importantly, however, the results point the way towards perceptual clustering and learning of new spatial concepts on real robots guided by task-related functional distinctions. Simulation-trained models may also be useful as a starting hypothesis for such learning. Altogether, the proposed method should be a useful tool for allowing robots to construct an understanding of space which will let them carry out meaningful tasks in our complex world.

A. Future work

Obviously, we would like to explore the proposed approach using a more realistic system, preferably a real robot. This will naturally entail a great deal of practical as well as theoretical problems, including sensory noise and imperfections, scene resetting and bootstrapping of the basic tasks.

All learning in this paper has been carried out in batch fashion; an online setting on a robot would benefit from a more goal-directed approach where only those micro-experiments were carried out for which the model is not yet certain.



Pair	LOC	FOR	SUP	PRO	CON
A,B	33 %	100 %	100 %	0 %	1 %
A,C	0 %	0 %	15 %	0 %	0 %
A,D	51 %	0 %	95 %	0 %	0 %
B,A	0 %	0 %	0 %	0 %	1 %
B,C	1 %	0 %	0 %	4 %	0 %
B,D	100 %	100 %	100 %	0 %	0 %
C,A	0 %	0 %	0 %	1 %	0 %
C,B	100 %	100 %	99 %	99 %	98 %
C,D	74 %	0 %	88 %	1 %	0 %
D,A	0 %	0 %	0 %	0 %	0 %
D,B	0 %	0 %	0 %	2 %	0 %
D,C	0 %	0 %	0 %	0 %	0 %

Fig. 4. Example scene evaluations using trained models.

Ultimately, the proposed approach should also be integrated into a larger framework, where there is learning of the low-level features themselves, as well as higher-level goals and drives that interact with the spatial concepts through tasks – and where the concepts form building blocks for higher level processes such as dialogue.

B. Acknowledgements

This work was supported by the SSF through its Centre for Autonomous Systems (CAS), as well as by the EU FP7 project CogX. We gratefully acknowledge the use of the SPARSEBAYES package by Mike Tipping (www.miketipping.com/), CGAL (www.cgal.org), BULLET (www.bulletphysics.com) and the Parma Polyhedra Library (www.cs.unipr.it/ppl).

Thanks are also extended to Dr. Carl Henrik Ek for much helpful input on the machine learning aspects.

REFERENCES

[1] Bullet physics library. http://www.bulletphysics.com.

[2] A. Aydemir, K. Sjöö, J. Folkesson, and P. Jensfelt. Search in the real world: Active visual object search based on spatial relations. In *IEEE International Conference on Robotics and Automation (ICRA)*, May 2011.

[3] I. Cos-Aguilera, L. Canamero, and G. Hayes. Using a sofm to learn object affordances. In *Proceedings of the 5th Workshop of Physical Agents (WAF04)*, 2004.

[4] K. Coventry and S. Garrod. *Saying, seeing and acting: the psychological semantics of spatial prepositions*. Hove, 2003.

[5] C. H. Ek, D. Song, K. Huebner, and D. Kragic. Exploring affordances in robot grasping through latent structure representation. In *Vision for Cognitive Tasks, ECCV*, September 2010.

[6] T. Flash and N. Hogan. The coordination of arm movements: an experimentally confirmed mathematical model. *Journal of Neuroscience*, 1985.

[7] J. Gibson. *The Ecological Approach to Visual Perception*. Lawrence Erlbaum Associates, Hillsdale, NJ., 1979.

[8] J. Kelleher. *A Perceptually Based Computational Framework for the Interpretation of Spatial Language*. PhD thesis, Dublin City University, 2003.

[9] J. Mugan and B. Kuipers. Continuous-domain reinforcement learning using a learned qualitative state representation. In *International Workshop on Qualitative Reasoning (QR-08)*, 2008.

[10] D. A. Norman. *The design of everyday things*. Doubleday, 1990.

[11] T. Regier. *The Human Semantic Potential*. MIT Press, 1996.

[12] 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.

[13] K. Sjöö, A. Aydemir, T. Mörwald, K. Zhou, and P. Jensfelt. Mechanical support as a spatial abstraction for mobile robots. In *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*, October 2010.

[14] D. Skočaj, G. Berginc, B. Ridge, A. Štimec, M. Jogan, O. Vanek, A. Leonardis, M. Hutter, and N. Hawes. A system for continuous learning of visual concepts. In *International Conference on Computer Vision Systems ICVS 2007*, 2007.

[15] M. E. Tipping and A. Faul. Fast marginal likelihood maximisation for sparse bayesian models. In *Proceedings of the Ninth International Workshop on Artificial Intelligence and Statistics*, Jan 2003.

[16] E. Uğur, M. R. Doğar, M. Çakmak, and E. Şahin. The learning and use of traversability affordance using range images on a mobile robot. In *2007 IEEE International Conference on Robotics and Automation*, April 2007.

[17] C. Vandeloise. *Spatial Prepositions: a case study from French*. The University of Chicago press, 1991.

[18] J. L. Wyatt, A. Aydemir, M. Brenner, M. Hanheide, N. Hawes, P. Jensfelt, M. Kristan, G.-J. M. Kruijff, P. Lison, A. Pronobis, K. Sjöö, A. Vrečko, H. Zender, M. Zillich, and D. Skočaj. Self-understanding & self-extension: a systems and representational approach. *IEEE Transactions on Autonomous Mental Development (TAMD), Special Issue on Representations and Architectures for Cognitive Systems*, Dec 2010.