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

DR 3.4:
Spatial knowledge and gaps therein

P. Jensfelt[1], H. Zender[2], C. Gretton[4], K. Sjöö[1], A. Aydemir[1], A. Pronobis[1], M. Göbelbacker[3], M. Hanheide[4], G.-J. Kruijff[2]

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

Due date of deliverable:	May 25, 2011
Actual submission date:	May 28, 2012
Lead partner:	KTH
Revision:	draft
Dissemination level:	PU

This deliverable deals with qualitative spatial cognition and in particular how to model knowledge about space and the gaps in this knowledge. Methods for how to identify and extend to fill these gaps are also presented. We present work in three directions. Firstly, we present an extension to the work on object search where we make full use of the switching planning framework developed in WP4. This allows us to, in a principled way, trade exploring the unknown versus exploiting the know part of space (i.e. perform object search). Second, we present additional results on the conceptual map and its incorporation into a full semantic mapping system. Finally, we present an analysis of a large database with indoor floor plans and some initial results on creating a model to predict the topology of space in an indoor environment.

Executive Summary

This report, DR.3.4, presents the work in WP3 which concerns the representation of space and how it is used to solve tasks. It describes the work during the fourth year of the CogX project on this topic and is accompanied by a second report, DR.3.3, from the same workpackage which focuses on how the spatial representation supports HRI and functional understanding. This report follow up on DR.3.1 and DR.3.2 from the previous years where we have presented the overall design of the spatial representation (task 3.1). We have also presented work on how qualitative spatial relations, like "in" and "on", can be used to represent space (task 3.2) and then especially long-term (task 3.3) and how it can be used to support HRI (task 3.4) and functional understanding (tasks 3.5).

During the first year we investigated the requirements for a spatial representation in a mobile cognitive system and started the design. We suggested a hierarchical representation with a metric mapping system at the lowest level to support traditional navigation tasks. Knowledge gaps on this level are mainly the location of the robot and the geometry of the unexplored space. To support high-level reasoning and planning we discretised space into places in the place layer. Gaps at this level corresponds to the unknown part of space, i.e. how to extend the place map. We also keep categorical models for objects, room types, etc in one additional layer. Finally, at the top we have the conceptual map which maintains a probabilistic model of high level concepts including the location of objects, segmentation of places into rooms, categorises of rooms, etc. During the second year we started the implementation of this representation with a focus on the lower levels and introduced object search as a benchmark task for the spatial representing. During the third year we showed how functional spatial relations can be used to abstract spatial knowledge, support reasoning and allow us to incorporate knowledge from online source where knowledge is often encoded in human readable form. We also presented a first version of the conceptual map.

During the fourth and final year we have continued to use object search as the motivating example for most of the work in this workpackage[1]. It provides a nice setting with a clear and relevant task which can be shown to benefit greatly from a deep understanding of space and its properties. We continued the work on starting from an unknown map and interleaving exploring the unknown and searching for objects in the known part of the world. The results of this is presented in [2].

In the work on object search we identified two common and important gaps in knowledge which we have studied further and present in this report. One of them is the knowledge about the category of rooms and places and

[1]Much of the work also in DR.3.3 has strong ties with object search

the other is knowledge about the part of space which we have not see yet. In DR.3.3 we present work on web mining for semantic scene understanding and object localization [20, 21].

As presented previously there is a strong correlation between the occurrence of object and the category of a room. For example, a frying pan is very strongly correlated with a kitchen and it is likely that even traveling quite far to find a kitchen is going to pay off versus searching non-kitchens that are closer. Being able to quickly and robustly categorise space is therefore a key capability. In [10, 11] we present our results.

In order to better predict the unknown part of space we have taken a learning approach. We have made use of and made significant contributions to building a large indoor floor plan database. This database consists of 197 buildings, 940 floors and over 38,000 rooms at the MIT and KTH campuses. We have analysed the data to look for exploitable patterns and have started learning models for the topology of indoor spaces [3, 4].

Role of spatial cognition in CogX

The overarching aim of CogX is to develop theories and methods for making robots able to self-understand and self-extend. One of the target platforms is a mobile robot moving around humans and in this context spatial knowledge is key. The robot needs to be able to deal with both large-scale and small-scale. It also needs to handle both what is currently in view and what is currently beyond the reach of the sensors as well as dealing with both short-term and long-term knowledge. This is what WP3 is about and it is therefore highly relevant for CogX.

Identifying gaps in knowledge is central in CogX as it is key to self-understanding and allows the robot to find plans for how to fill those gaps and thus self-extend. In WP3 we work with gaps in spatial knowledge, such as the category of a room, the location of an object, and what lies ahead in the yet unexplored part of space.

Contribution to the CogX scenarios and prototypes

The work in WP3 is foundational for the Dora demonstrator that is all about acting in space. It is also a requirement for much of the work on situated dialogue processing (WP6) and there is a strong interaction with the planning workpackage.

1 Tasks, objectives, results

1.1 Planned work

Qualitative spatial cognition is the topic of WP3. In this project we focus on higher-level tasks rather than the traditional navigation tasks often associated with space. That said we still need to support such functionality. However, the main thrust is on endowing the robot with the ability to go well beyond that and reason about space, talk about space, etc. We want to support natural interaction between humans and robots. By this we mean that humans should not have to be trained to speak the language of the robot but rather the robot should be able to reason in terms of human concepts. During the course of the project we have learned that this has the additional benefit that the robot is able to make use of the plethora of spatial knowledge encoded in human readable form on the Internet.

We are interested in representing not only the belief about things we know about but also beliefs about the beliefs. That is, we want to endow the robot with the ability to reason about the extent of its own knowledge. For example, this means that the robot needs to not only know its own position in space but also how well its knows it and, very importantly, if it does not know. Also, it is not enough to have a model for what we know about the space around us, we also need to represent what we do not know. By representing these gaps in knowledge we give the robot a certain level of self-understanding. This self-understanding provides the means by which the system can plan to self-extend which is the goal in CogX. This report presents the work from the final year on finding, representing and planning to fill gaps in knowledge, in this specific case in the context of spatial knowledge. This corresponds to task 3.6.

Task 3.6: Representations of gaps in spatial knowledge. *How to represent beliefs about beliefs of spatial knowledge.*

The plan from the start of the project was study methods for self-understanding and self-extension and task 3.6 represents this work in the context of spatial cognition. The aim during the final year was to finalise the work on the conceptual map and make full use of it. We planned to apply the planning techniques developed in WP4 to the problem of object search to showcase planning for filling gaps in spatial knowledge and doing so in a setup where not only the location of the object is unknown but the layout of space in general. The scenario is that the robot comes to an unknown environment, equipped only with its categorical knowledge, and has to find a certain object. In this scenario there are two competing subtasks, one is to explore space and the other is to look for the object in the explored part of space. Initially the only available option is to explore but as soon as some part of space is know the robot needs to decide if it is time to *exploit* its

current knowledge and look for the object or *explore* more of the unknown space.

The work in this deliverable directly contributes to the following project objectives.

O2 Specific representations of beliefs about beliefs for the specific cases of dialogue, manipulation, maps, mobility and some types of vision.

O3 Representations of how actions will alter the belief state of the cognitive system, and those of other agents, as represented in the first two objectives, i.e. models of the effects of actions on beliefs about space, categorical knowledge, action effects, dialogue moves etc.

O7 Methods for perception and spatial modelling that enable a robot to identify gaps in its spatial models (e.g. maps) and to extend them so as to support natural communication with humans.

O11 A robotic implementation of our theory able to complete a task involving mobility, interaction and manipulation, in the face of novelty, uncertainty, partial task specification, and incomplete knowledge.

The research in WP3 has also contributed greatly to meeting the following objective through the tight connection between the spatial representation and the planner (WP4) in the work on active object search

O4 A theory of how to reason, plan, act and interact using such representations of beliefs, and beliefs about beliefs, to achieve a task in the face of incomplete information, uncertainty and novelty.

1.2 Actual work performed

1.2.1 Task 3.6: Representations of gaps in spatial knowledge.

We continued the work on object search as planned and made use of the switching planner framework developed in WP4 to create, execute and monitor plans to fill gaps in spatial knowledge. This was reported in [2] (annex 2.1) and makes full use of the conceptual map and allowed the system to start from a completely unknown map and trade exploration versus exploitation (i.e. searching for the object in the known part of space). The final part of the implementation the conceptual map and experiments and evaluations were performed and reported in [10, 11] (annexes 2.2 and 2.3). The most explicit example of how we model and reason about gaps in knowledge is our way of reasoning about possible extensions to the perceived part of the world. We create hypothetical worlds and use the inference mechanisms of the conceptual map to estimate the cost and likelihood of finding the object in the unexplored part of space. Using the planner allowed us to address the problem of trading exploitation versus exploration in a principled way,

without following a certain hard-coded strategy. We use the planner to find the strategy given the current belief model. The belief model evolves as the robot moves around and gathers information.

As predicting what lies ahead in the unexplored part of space was shown to be a key capability and we expanded the work on in this direction. We made use of a large dataset of floor plans from indoor environments to learn priors for the topology of indoor spaces. This was reported in [3, 4] (annexes 2.4 and 2.5). This is to our knowledge the first time anyone makes use of such a large database over indoor environments. We believe that this will be an important step towards a better understanding of indoor environments and thereby more efficient robots.

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 3.6

Object search has gained more and more interest in the literature recently which we believe to be because it is a challenging task that can benefit greatly from semantic information and careful planning. Brute force strategies are prohibitively slow. Tsotsos, Ye and colleagues present methods for computing the next best view to move the camera to in an object search task [19, 15]. [1] employs a similar strategy to object search in a 4x4 meters room, i.e., a quite small environment. [7] describes a mobile robot that autonomously locates objects in an entire floor of an office building. The map of the environment is fully known *a priori*. Furthermore, the system extracts object-object co-occurrence probabilities from an annotated image database. The biggest limitation of the system is the assumption of a known map and previously detected objects throughout the whole environment. [16] presents a similar system in which a method for place labelling is used to bootstrap the search. As [7] this approach also uses the semantics of the environment to make the search more efficient. In [6] a POMDP planner is used for object search with single or multiple searchers. The authors provide simulation results and a proof-of-concept implementation where a mobile robot is tasked to find cups in an already known environment with known search positions that the robot may choose to stop and take a picture from. Our work goes beyond what is found in the literature in several ways. We do not require the map to be known in advance, or as in some other work require the system to first explore the entire environment and we operate in a large-scale environment. We can also reason about the unexplored space and we can trade exploration versus exploitation without having to rely on hard-coded strategies. Our system adapts its behaviour based on the cur-

rent beliefs. In the process of filling the gap in knowledge about the location of the object in question it exploits the spatial representation and actively plans to fill other gaps that are found to help on the way. An example of this, shown during the third year, is when the system plans to find a table before it looks for the book because it estimates the cost of finding the table and then looking for the book on the table to be lower than looking for the book directly. It is worth pointing out that the same planning framework is able to handle many other types of goals in addition to finding objects. Given all of this, we clearly address the problem of object search in a more principled way than in previous work.

There is a relatively rich literature on place recognition and lately also on categorisation. Examples from computer vision include the work by Torralba *et.al.* [14, 13]. Early work from robotics include Buschka & Saffiotti [5] and Mozos *et.al.* [8]. The latter applies boosting to create a classifier based on a set of geometrical features extracted from range data to classify different places in indoor environments into rooms, corridors and doorways. Viswanathan *et.al.* [17] build their system on objects and perform automated learning of object-place relations and visual object models from the online LabelMe database. In [18] a Bayesian filtering scheme is added on top of the frame-based categorisation to increase robustness and give a smoother category estimate. Most recently, Ranganathan [12] casts the problem in a fully probabilistic framework which operates on sequences of images rather than individual images. The method uses change point detection to detect abrupt changes in the statistical properties of the data. Our work on the conceptual map differs from the literature in that we have a fully probabilistic approach from common-sense to semantic mapping and embed the place categorisation into the system where it operates.

Finally, in terms of learning models of topology of indoor spaces there is relatively little prior work. In [9], which deals with place categorisation, a HMM is added 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. This model only gives the probability to transition from one type to another in general. In our work we instead try to answer this question conditioned on the topology know so far. We also believe that we are the first to make use of such a huge database to analyse indoor spaces.

2 Annexes

2.1 A. Aydemir et. al. "Plan-based object search and exploration using semantic spatial knowledge in the real world"

Bibliography A. Aydemir, M. Gbelbecker, A. Pronobis, K. Sj, and P. Jensfelt. "Plan-based object search and exploration using semantic spatial knowledge in the real world". In Proceedings of the 5th European Conference on Mobile Robots (ECMR'11), Örebro, Sweden, Sept. 2011.

Abstract In this paper we present a principled planner based approach to the active visual object search problem in unknown environments. We make use of a hierarchical planner that combines the strength of decision theory and heuristics. Furthermore, our object search approach leverages on the conceptual spatial knowledge in the form of object cooccurences and semantic place categorisation. A hierarchical model for representing object locations is presented with which the planner is able to perform indirect search. Finally we present real world experiments to show the feasibility of the approach.

Relation to WP This paper represents one of the most central integrated results of self-understanding and self-extension in terms of space. The system models its knowledge and lack of knowledge and plans to fill these gaps by either learning more about space or searching specifically for a certain object.

2.2 A. Pronobis and P. Jensfelt. "Hierarchical multi-modal place categorization"

Bibliography A. Pronobis and P. Jensfelt. "Hierarchical multi-modal place categorization". In Proceedings of the 5th European Conference on Mobile Robots (ECMR'11), Örebro, Sweden, Sept. 2011.

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 presents the place categorization system used in our system. The category of places is one of the most important pieces of knowledge or lack there of in case of a gap for the system when performing human type tasks.

2.3 A. Pronobis and P. Jensfelt. "Large-scale semantic mapping and reasoning with heterogeneous modalities"

Bibliography A. Pronobis and P. Jensfelt. "Large-scale semantic mapping and reasoning with heterogeneous modalities". In Proceedings of the 2012 IEEE International Conference on Robotics and Automation (ICRA'12), Saint Paul, MN, USA, May 2012.

Abstract This paper presents a probabilistic framework combining heterogeneous, uncertain, information such as object observations, shape, size, appearance of rooms and human input for semantic mapping. It abstracts multi-modal sensory information and integrates it with conceptual commonsense knowledge in a fully probabilistic fashion. It relies on the concept of spatial properties which make the semantic map more descriptive, and the system more scalable and better adapted for human interaction. A probabilistic graphical model, a chain-graph, is used to represent the conceptual information and perform spatial reasoning. Experimental results from online system tests in a large unstructured office environment highlight the system's ability to infer semantic room categories, predict existence of objects and values of other spatial properties as well as reason about unexplored space.

Relation to WP This paper presents the semantic mapping system as a whole and an evaluation of it. To some extent this represents the implementation of the spatial self-understanding and it is what allows the system to plan to self-extend.

2.4 A. Aydemir et. al., "Predicting indoor topology labelings and structure from a large indoor topological database"

Bibliography A. Aydemir, E. Järleberg, S. Prentice and P. Jensfelt, "Predicting indoor topology labelings and structure from a large indoor topological database", Spatial Cognition, 2012

Abstract A signi

cant amount of research in robotics is aimed towards building robots that operate indoors yet there exists little analysis of how human spaces are organized. In this work we analyze the properties of indoor environments from a large annotated oorplan dataset. We analyze a corpus of 567 oors, 6426 spaces with 91 room types and 8446 connections between rooms corresponding to real places. We present a system that, given a partial graph, predicts the rest of the topology by building a model from this dataset. Our hypothesis is that indoor topologies consists of multiple smaller functional parts. We demonstrate the applicability of our approach with experimental results. We expect that our analysis paves the way for more data driven research on indoor environments.

Relation to WP The gap in knowledge in terms of what lies in the unknown part of space has been shown in the work on object search to be very important. This paper presents an initial analysis of a large indoor dataset of floor plans to shed some light on how to learn models for this.

2.5 A. Aydemir et. al., "What can we learn from 38,000 rooms? Reasoning about unexplored space in indoor environments"

Bibliography A. Aydemir, P. Jensfelt and J. Folkesson, "What can we learn from 38,000 rooms? Reasoning about unexplored space in indoor environments", submitted to the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2012

Abstract Many robotics tasks require the robot to predict what lies in the unexplored part of the environment. In this paper, we propose and compare two methods for predicting both the topology and the categories of rooms given a partial map. The methods are motivated by the analysis of two large annotated floor plan data sets corresponding to the buildings of the MIT and KTH campuses. In particular, utilizing graph theory, we discover that local complexity remains unchanged for growing global complexity in real-world indoor environments, a property which we exploit. In total, we analyze 197 buildings, 940 floors and over 38,000 real-world rooms. Such a large set of indoor places has not been investigated before in the previous work. We provide extensive experimental results and show the degree of transferability of spatial knowledge between two geographically distinct locations. We also contribute the KTH data set and the software tools to work with it.

Relation to WP This paper presents a deeper analysis of a large indoor dataset of floorplans gathered at the MIT and KTH campuses, containing more than 38,000 rooms. The paper also presents a model for how to model the gap in spatial knowledge when faced with unexplored space.

References

[1] A. Andreopoulos, S. Hasler, H. Wersing, H. Janssen, J.K. Tsotsos, and E. Korner. Active 3d object localization using a humanoid robot. *Robotics, IEEE Transactions on*, 27(1):47 –64, feb. 2011.

[2] 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 *Proc. of the European Conference on Mobile Robotics (ECMR'11)*, Örebro, Sweden, September 2011.

[3] Alper Aydemir, Erik Järleberg, Samuel Prentice, and Patric Jensfelt. Predicting indoor topology labelings and structure from a large indoor topological database. In *Proc. of Spatial Cognition*, 2012.

[4] Alper Aydemir and Patric Jensfelt. Exploiting and modeling local 3d structure for predicting object locations. In *submitted to Proc. of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'12)*, 2012.

[5] Pär Buschka and Alessandro Saffiotti. A virtual sensor for room detection. In *Proceedings of the 2002 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'02)*, Lausanne, Switzerland, 2002.

[6] Geoffrey Hollinger, Dave Ferguson, Siddhartha Srinivasa, and Sanjiv Singh. Combining search and action for mobile robots. In *ICRA'09: Proceedings of the 2009 IEEE international conference on Robotics and Automation*, pages 800–805, Piscataway, NJ, USA, 2009. IEEE Press.

[7] Thomas Kollar and Nicholas 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*, pages 4116–4121, Piscataway, NJ, USA, 2009. IEEE Press.

[8] 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.

[9] 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.

[10] Andrzej Pronobis and Patric Jensfelt. Hierarchical multi-modal place categorization. In *Proc. of the European Conference on Mobile Robotics (ECMR)*, Örebro, Sweden, September 2011.

[11] Andrzej Pronobis and Patric Jensfelt. Large-scale semantic mapping and reasoning with heterogeneous modalities. In *Proceedings of the 2012 IEEE International Conference on Robotics and Automation (ICRA'12)*, Saint Paul, MN, USA, May 2012.

[12] 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.

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

[14] 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.

[15] John K. Tsotsos and Ksenia Shubina. Attention and visual search : Active robotic vision systems that search. 2007.

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

[17] 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.

[18] 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.

[19] Yiming. Ye. *Sensor planning for object search*. PhD thesis, 1998.

[20] Kai Zhou, Karthik Mahesh Varadarajan, Michael Zillich, and Markus Vincze. Web mining driven semantic scene understanding and object localization. In *IEEE International Conference on Robotics and Biomimetics (ROBIO)*, Phuket, Thailand, Dec 2011.

[21] Kai Zhou, Michael Zillich, and Markus Vincze. Web mining driven object locality knowledge acquisition for efficient robot behavior. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (submitted)*, Vilamoura, Algarve, Portugal, Oct 2012.

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. Support 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.

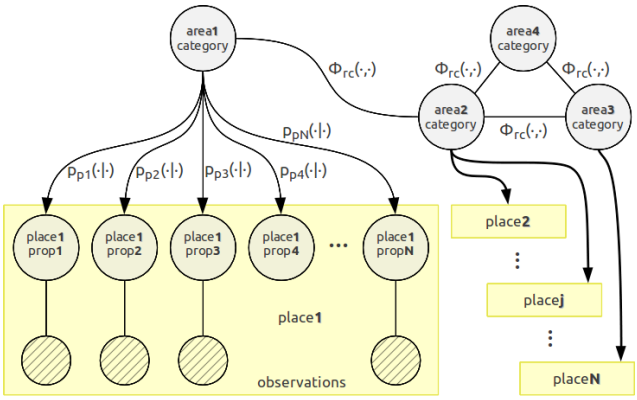


Fig. 1. Structure of the graphical model for the places showing the influence of the properties and the topology on the categorization and segmentation.

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 re-evaluating 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

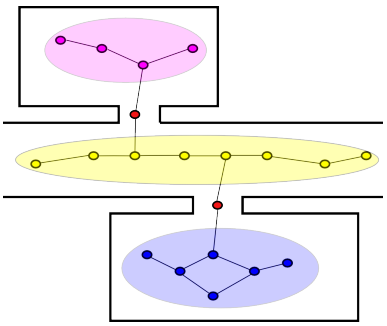


Fig. 2. The set of places, $\{p_i\}$, is segmented into areas based on the door places. The doors form the edges in the topological area graph.

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).

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 were not found this way were assigned a certain low default probability not to rule them out completely.

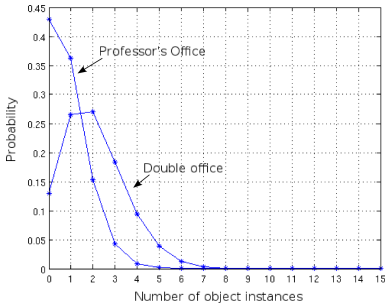


Fig. 3. The Poisson distributions modelling the existence of a certain number of objects in a room on the example of computers present in a double office and a professor's office.

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.

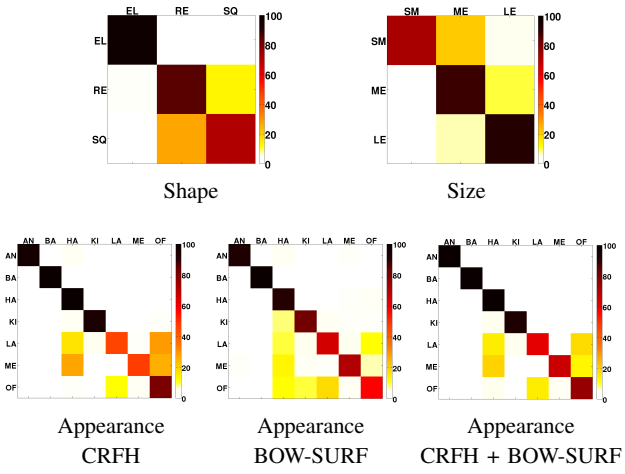


Fig. 4. Confusion matrices for the evaluation of the property categorizers.

C. Real-world experiments

In the real-world experiment the robot was manually driven through the environment using a joystick. 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.

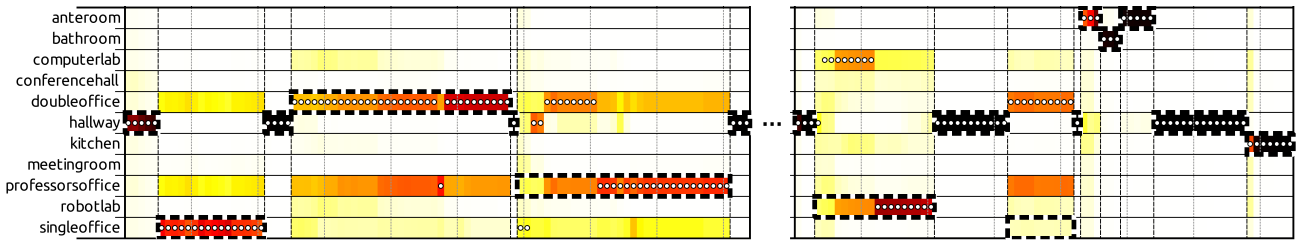


Fig. 5. Visualization of the beliefs about the categories of the rooms. The room category ground truth is marked with thick dashed lines while the MAP value is indicated with white dots.

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.

Plan-based Object Search and Exploration Using Semantic Spatial Knowledge in the Real World

Alper Aydemir* Moritz Göbelbecker† Andrzej Pronobis* Kristoffer Sjöö * Patric Jensfelt*
*Centre for Autonomous Systems, Royal Institute of Technology, Stockholm, Sweden
†Institut für Informatik, Albert-Ludwigs-Universität Freiburg, Germany

***Abstract*— In this paper we present a principled planner based approach to the active visual object search problem in unknown environments. We make use of a hierarchical planner that combines the strength of decision theory and heuristics. Furthermore, our object search approach leverages on the conceptual spatial knowledge in the form of object cooccurences and semantic place categorisation. A hierarchical model for representing object locations is presented with which the planner is able to perform indirect search. Finally we present real world experiments to show the feasibility of the approach.**

***Index Terms*— Active Sensing, Object Search, Semantic Mapping, Planning**

I. Introduction

Objects play an important role when building a semantic representation and an understanding of the function of space [14]. Key tasks for service robots, such as fetch-and-carry, require a robot to successfully find objects. It is evident that such a system cannot rely on the assumption that all object relevant to the current task are already present in its sensory range. It has to actively change its sensor parameters to bring the target object in its field of view. We call this problem *active visual search* (AVS).

Although researchers began working on the problem of visually finding a relatively small sized object in a large environment as early as 1976 at SRI [4], the issue is often overlooked in the field. A common stated reason for this is that the underlying problems such as reliable object recognition and mapping are posing hard enough challenges. However as the field furthers in its aim to build robots acting in realistic environments, this assumption need to be relaxed. The main contribution of this work a method to relinquish the above mentioned assumption.

A. Problem Statement

We define the active visual object search problem as an agent localizing an object in a known or unknown 3D environment by executing a series of actions with the lowest total cost. The cost function is often defined as the time it takes to complete the task or distance traveled.

Let the environment be Ω and Ψ being the search space with $\Psi \subseteq \Omega$. Also let $P_o(\Psi)$ be the probability distribution for the position of the center of the target object o defined as a function over Ψ. The agent can execute a sensing action s in the reachable space of Ψ. In the case of a camera as the sensor, s is characterised by the camera position, (x_c, y_c, z_c), pan-tilt angles (p, t), focal length f and a recognition algorithm a; $s = s(x_c, y_c, z_c, p, t, f, a)$. The part of Ψ covered by s is called a *viewcone*. In practice, a has an effective region in which reliable recognition or detection is achieved. For the i^{th} viewcone we call this region V_i.

Depending on the agent's level of a priori knowledge of Ψ and $P_o(\Psi)$ there are three extreme cases of the AVS problem. If both Ψ and $P_o(\Psi)$ is fully known then the problem is that of sensor placement and coverage maximization given limited field of view and cost constraints.

If both Ψ and $P_o(\Psi)$ is unknown then the agent has an additional *explore* action as well. An exhaustive exploration strategy is not always optimal, i.e. the agent needs to select which parts of the environment to explore first depending on the target object's properties. Furthermore the agent needs to trade-off between executing a sensing action and exploration at any given point. That is, should the robot search for the object o in the partially known Ψ or explore further. This is classically known as the exploration vs. exploitation problem.

When $P_o(\Psi)$ is unknown (i.e. uniformly distributed) but Ψ is known (i.e. acquired a priori), the agent needs to gather information about the environment similar to the above case. However in this case, the exploration is for learning about the target object specific characteristics of the environment. Knowing Ψ also means that the robot can reason whether or not to execute a costly search action at the current position, or move to another more promising region of space. The rare case where $P_o(\Psi)$ is fully known but Ψ is unknown is not practically interesting to the scope of this paper.

So far, we have examined the case where the target object is an instance. The implication of this is that $P_o(\Psi) + P_o(\Omega \setminus \Psi) = 1$, therefore observing V_i has an effect on $P_o(\Psi \setminus V_i)$. However this is not necessarily true if instead the agent is searching for any member of an object category and the number of them is not known in advance. Therefore knowing whether the target object is a unique instance or a member of an object category is an important factor in search behavior.

Recently there's an increasing amount of work on acquiring *semantic maps*. Semantic maps have parts of the environment labeled representing various high level concepts and functions of space. Exploring and building a semantic map while performing AVS contributes to the estimation of $P_o(\Psi)$. The semantic map provides information that can be exploited by leveraging on common-sense conceptual knowledge about

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

indoor environments. This knowledge describes, for example, how likely it is that plates are found in kitchens, that a mouse and a computer keyboard occur in the same scene and that corridors typically connect multiple rooms. Such information offers valuable information in limiting the search space. The sources for those can be from online common-sense databases or world wide web among others. Acknowledging the need to limit the search space and integrate various cues to guide the search, [4] proposed *indirect search*. Indirect search as a search strategy is a simple and powerful idea: it's to find another object first and then use it to facilitate finding the target object, e.g. finding a table first while looking for a landline phone. Tsotsos [13] approached the problem by analyzing the complexity of the AVS problem and showed that it is NP-hard. Therefore we must adhere to a heuristics based solution. Ye [15] formulated the problem in probabilistic framework.

In this work we consider the case where Ψ and $P_o(\Psi)$ are both unknown. However, the robot is given probabilistic default knowledge about the relation betweeen objects and the occurences of objects in difference room category following our previous work [1, 6].

B. Contributions

The contributions of this work are four fold. First we provide the domain adaptation of a hierarchical planner to address the AVS problem. Second we show how to combine semantic cues to guide the object search process in a more complex and larger environment than found in previous work. Third, we start with an unknown map of the environment and provide an exploration strategy which takes into account the object search task. Four, we present real world experiments searching for multiple objects in a large office environment, and show how the planner adapts the search behavior depending of the current conditions.

C. Outline

The outline of this paper is as follows. First we present how the AVS problem can be formulated in a principled way using a planning approach (Section II). Section III provides the motivation for and structure of various aspects of our spatial representation. Finally we showcase the feasibility of our approach in real world experiments (Section IV).

II. Planning

For a problem like AVS which entails probabilistic action outcomes and world state, the robot needs to employ a planner to generate flexible and intelligent search behavior that trade off exploitation versus exploration. In order to guarantee optimality a POMDP planner can be used in, i.e. a decision theoretic planner that can accurately trade different costs against each other and generate the optimal policy. However, this is only tractable when a complex problem like AVS is applied to very small environments. Another type of planner are the classical AI planners which requires perfect knowledge about the environment. This is not the case since both Ψ and $P_o(\Psi)$ are unknown.

A variation of the classical planners are the so called continual planners that interleave planning and plan monitoring in order to deal with uncertain or dynamic environments[3]. The basic idea behind the approach is to create an plan that *might* reach the goal and to start executing that plan. This initial plan takes into account success probabilities and action costs however it is optimistic in nature. A monitoring component keeps track of the execution outcome and notifies the planner in the event of the current plan becoming invalid (either because the preconditions of an action are no longer satisfied or the plan does not reach the goal anymore). In this case, a new plan is created with the updated current state as the initial state and execution starts again. This will continue until either the monitoring component detects that the goal has been reached or no plan can be found anymore.

In this paper we will make use of a so called switching planner. It combines two different domain independent planners for different parts of the task: A *classical continual planner* to decide the overall strategy of the search (for which objects to search in which location) and a *decision theoretic planner* to schedule the low level observation actions using a probabilistic sensing model. Both planners use the same planning model and are tightly integrated.

We first give a brief description of the switching planner. We focus on the use of the planner in this paper and instead refer the reader to [5] for a more detailed description. We will also present the domain modeling for the planner, and give further details on various aspects of knowledge that planner makes use of.

A. Switching Planner

1) Continual Planner (CP): We build our planning framework on an extended SAS^+[2] formalism. As a base for the continual planner, we use Fast Downward[7]. Because our knowledge of the world and the effects of our actions are uncertain we associate a *success probability* $\mathrm{p}(a)$ with every action a. In contrast to more expressive models like MDPs or even POMDPs, actions don't have multiple possible outcomes, they just can succeed with probability $\mathrm{p}(a)$ or fail with probability of $1 - \mathrm{p}(a)$.

The goal of the planner is then to find a plan π that reaches the goal with a low cost. In classical planning the cost function is usually either the number of actions in a plan or the sum of all action's costs. Here we chose a function that resembles the expected reward adjusted to our restricted planning model. With $\mathrm{p}(\pi) = \prod_{a\in\pi} \mathrm{p}(a)$ as the plans total success probability and $\mathrm{cost}(\pi) = \sum_{a\in\pi} \mathrm{cost}(a)$ as the total costs, we get for the optimal plan π^*:

$$\pi^* = \underset{\pi}{\operatorname{argmin}}\, \mathrm{cost}(\pi) + R(1 - \mathrm{p}(\pi))$$

where a is an action and the constant R is the reward the planner is given for achieving the goal. For small values of R the planner will prefer cheaper but more unlikely plans, for larger values more expensive plans will be considered.

Assumptions The defining feature of an exploration problem is that the world's state is uncertain. Some planning frameworks such as MDPs allow the specification of an initial state distribution. We choose not to do this for two different reasons: a) having state distributions would be a too strong departure from the classical planning model and b) the typical exploration problems we deal with have too many possible

states to express explicitly. We therefore use an approach we call *assumptive actions* that allow the planner to construct parts of the initial state on the fly, and which allows us to map the spatial concepts to the planning problem in an easy way.

2) Decision Theoretic (DT) Planner: When the continual planner reaches a sensing action (e.g. *search location1 for a object2*), we create a POMDP that only contains the parts of the state that are relevant for that subproblem with. This planner can only use MOVE and PROCESSVIEWCONE actions explained in Section II-B.2. The DT planner operates in a closed-loop manner, sending actions to be executed and receiving observations from the system. Once the DT planner either confirms or rejects a hypothesis, it returns control back to the continual planner, which treats the outcome of the DT session like the outcome of any other action.

B. Domain Modeling

We need to discretize the involved spaces (object location, spatial model and actions) to make a planner approach applicable to the AVS problem. Most methods make use of discretizations as a way to handle the NP-hard nature of the problem.

1) Representing space: For the purposes of obstacle avoidance, navigation and sensing action calculation, Ψ is represented as a 3D metric map. Ψ discretised into i volumetric cells so that $\Psi = c_0...c_i$. Each cell represents the occupancy with the attributes OCCUPIED, FREE or UNKOWN as well as the probability of target object's center being in that cell.

However, further abstraction is needed to achieve reliable and fast plan calculation as the number of cells can be high. For this purpose we employ a topological representation of Ψ called *place map*, see Fig 1(a). In the place map, the world is represented by a finite number of basic spatial entities called *places* created at equal intervals as the robot moves. Places are connected using paths which are discovered by traversing the space between places. Together, places and paths represent the topology of the environment. This abstraction is also useful for a planner since metric space would result in a largely intractable planning state space.

The places in the place map are grouped into rooms. In the case of indoor environments, rooms are usually separated by doors or other narrow openings. Thus, we propose to use a door detector and perform reasoning about the segmentation of space into rooms based on the doorway hypotheses. We use a template-based door detection algorithm which matches a door template to each acquired laser scan. This creates door hypotheses which are further verified by the robot passing through a narrow opening.

In addition, unexplored space is represented in the place map using hypothetical places called *placeholders* defined in the boundary between free and unknown space in the metric map.

We represent object locations not in metric coordinates but in relation to other known objects or rooms to achieve further abstraction. The search space is considered to be divided into *locations* $\mathcal{L}$. A location is either a *room* $\mathcal{R}$ or a *related space*. Related spaces are regions connected with a *landmark object* o, either *in* or *on* the landmark (see [1] for more details). The related space "in" o is termed $\mathcal{I}_o$ and the space "on" o $\mathcal{O}_o$.

2) Modeling actions: The planner has access to three physical actions: MOVE can be used to move to a place or placeholder, CREATEVIEWCONES creates sensing actions for an *object label* in *relation* to a specified *location*, PROCESSVIEWCONE executes a sensing action. Finally, the virtual SEARCHFOROBJECT action that triggers the decision theoretic planner.

3) Virtual objects: There are two aspects of exploration in the planning task: we're searching for an (at that moment) unknown object, which may include the search for support objects as an intermediate step. But the planner may also need to consider the utility of exploring its environment in order to find new rooms in which finding the goal object is more likely.

Because the planners we use employ the closed world assumption, adding new objects as part of the plan is impossible. We therefore add a set of *virtual objects* to the planning problem that can be instantiated by the planner as required by the plan. This approach will fail for plans that require finding more objects than pre-allocated, but this is not a problem in practice. The monitoring component tries to match new (real) objects to virtual objects that occur in the plan. This allows us to deliver the correct observations to the DT planner and avoid unnecessary replanning.

4) Probabilitic spatial knowledge: The planner makes use of the following probabilistic spatial knowledge in order to generate sensible plans:

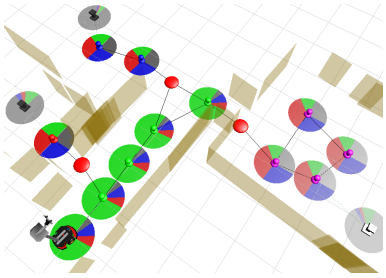
- $P_{category}(room_i)$ defines the distribution over room categories that the robot has a model for, for a given room integrated over places that belongs to $room_i$. The planner uses this information to decide whether to plan for a SEARCHFOROBJECT action or explore the remaining placeholders.
- $P_{category}(placeholder_i)$ represents the probability distribution of a placeholder turning into a new room of a certain category upon exploration. Using this distribution, the planner can choose to explore a placeholder instead of another, or plan to launch search altogether.
- $P(ObjectAt\mathcal{L})$ gives the probability of an object o being at location $\mathcal{L}$.

More details about calculation of these probabilities are further explained in Section III.

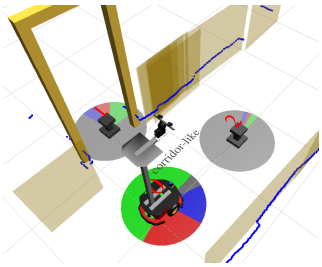
III. SPATIAL REPRESENTATION

5) Conceptual Map: All higher level inference is performed in the so called conceptual map which is represented by a graphical model. It integrates the conceptual knowledge (food items are typically found in kitchens) with instance knowledge (the rice package is in room4). We model this in a *chain graph* [8], whose structure is adapted online according to the state of underlying topological map. Chain graphs provide a natural generalisation of directed (Bayesian Networks) and undirected (Markov Random Fields) graphical models, allowing us to model both "directed" causal as well as "undirected" symmetric or associative relations.

The structure of the chain graph model is presented in Fig. 2. Each discrete place is represented by a set of random variables connected to variables representing semantic category of a room. Moreover, the room category variables are connected



(a)



(b)

Fig. 1. (a) A place map with several places and 3 detected doors shown as red. (b) Shows two placeholders with different probabilities for turning into new rooms: one of them is behind a door hypothesis therefore having a higher probability of leading into a new room. Colors on circular discs indicates the probability of room categories as in a pie chart: i.e. the bigger the color is the higher the probability. Here green is *corridor*, red is *kitchen* and blue is *office*.

by undirected links to one another according to the topology of the environment. The potential functions $\phi_{rc}(\cdot,\cdot)$ represent the type knowledge about the connectivity of rooms of certain semantic categories.

To compute $P_{category}(room_i)$ each place is described by a set of properties such as size, shape and appearance of space. These are are based on sensory information as proposed in [12]. We extend this work by also including presence of a certain number of instances of objects as observed from each place as a properties (due to space limitations we refer to [11] for more details). This way object presence or absence in a room also affects the room category. The property variables can be connected to observations of features extracted directly from the sensory input. Finally, the functions $p_s(\cdot|\cdot)$, $p_a(\cdot|\cdot)$, $p_{o_i}(\cdot|\cdot)$ utilise the common sense knowledge about object, spatial property and room category co-occurrence to allow for reasoning about other properties and room categories.

For planning, the chain graph is the sole source of belief-state information. In the chain graph, belief updates are event-driven. For example, if an appearance property, or object detection, alters the probability of a relation, inference proceeds to propagate the consequences throughout the graph. In our work, the underlying inference is approximate, and uses the fast Loopy Belief Propagation [9] procedure.

A. Object existence probabilities

To compute the $P(ObjectAt\mathcal{L})$ value used in active visual search in this paper, objects are considered to be occurring:

1) independently in different locations $\mathcal{L}$
2) independently of other objects in the same location

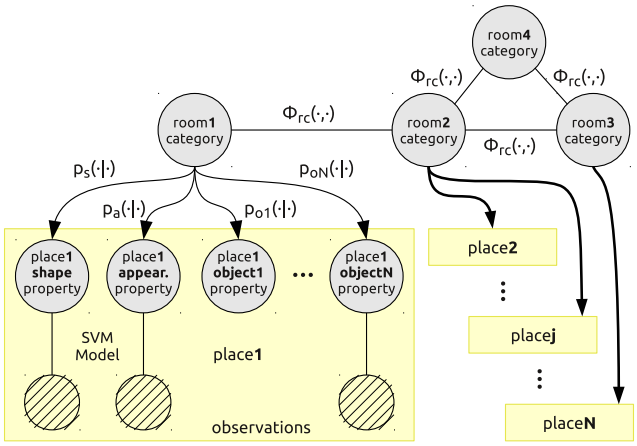


Fig. 2. Schematic image of chain graph

3) as Poisson processes over cells $c_0 ... c_i$ per location $\mathcal{L}$

In other words, each location has the possibility of containing, independently of all other locations, a number n_c of objects of a class c with probability

$$P(n_c = k) = \frac{\lambda^{k}_{\mathcal{L},c} e^{-\lambda_{\mathcal{L},c}}}{\mathrm{k!}} \tag{1}$$

where $\lambda_{\mathcal{L},c}$ is the expected number of objects of class c in the location $\mathcal{L}$. The probability of *at least one* object in a location is

$$P(n_c > 0) = 1 - P(n_c = 0) = 1 - e^{-\lambda_{i,c}} \tag{2}$$

Because of the independence assumptions, the λ values for a location and all its subordinate locations can simply be added together to obtain the distribution of the number of objects of that class occurring in that whole hierarchy.

1) Exploration: In addition to making inferences about explored space, the conceptual map can provide predictions about unexplored space. To this end, we extend the graph by including the existence of placeholders. For each placeholder a set of probabilities is generated that the placeholder will lead to a room of a certain category.

This process is repeated for each placeholder and consists of three steps. In the first step, a set of hypotheses about the structure of the unexplored space is generated. In case of our implementation, we evaluate 6 hypotheses: (1) placeholder does not lead to new places, (2) placeholder leads to new places which do not lead to a new room, (3) placeholder leads to places that lead to a single new room (4) placeholder leads to places that lead a room which is further connected to another room, (5) placeholder leads to a single new room directly, and (6) placeholder leads to a new room directly which leads to another room. In the second step, the hypothesized rooms are added to the chain graph just like regular rooms and inference about their categories is performed. Then, the probability of any of the hypothesized rooms being of a certain category is obtained. Finally, this probability is multiplied by the likelihood of occurrence of each of the hypothesized worlds estimated based on the amount of open space behind the placeholder and the proximity of gateways. A simple example is shown in Fig. 1(b)

IV. Experiments

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 12x8 m environment with 3 different rooms, *kitchen*, *office1*, *office2* connected by a corridor. The robot had models of all objects it searches for before each search run. 3 different objects (*cerealbox*, *stapler* and *whiteboardmarkers*) were used during experiments. The BLORT framework was used to detect objects [10].

To highlight the flexibility of the planning framework evaluated the system with 6 different starting positions and tasked with finding different objects in an unknown environment. We refer the reader to `http://www.csc.kth.se/~aydemir/avs.html` for videos. Each sub-figure in Fig. 3 shows the trajectory of the robot. The color coded trajectory indicates the room category as perceived by the robot: red is kitchen, green is corridor and blue is office. The two green arrows denote the current position and the start position of the robot.

In the following we give a brief explanation for what happened in the different runs.

- Fig. 3(a) Starts: *corridor*, Target: *cerealbox* in *kitchen*
  The robot starts by exploring the *corridor*. The robot finds a doorway on its left and the placeholder behind it has a higher probability of yielding into a kitchen and the robot enters *office1*. As the robot acquires new observations the CP's kitchen assumption is violated. The robot returns to exploring the corridor until it finds the kitchen door. Here the CP's assumptions are validated and the robot searches this room. The DT planner plans a strategy of first finding a table and then the target object on it. After finding a table, the robot generates view cones for the $\mathcal{O}_{table,cornflakes}$ location. The cerealbox object is found.
- Fig. 3(b) Starts: *office2*, Target: *cerealbox* in *kitchen*
  Unsatisfied with the current room's category, the CP commits to the assumption that exploring placeholders in the corridor will result in a room with category kitchen. The rest proceeds as in Fig. 3(a).
- Fig. 3(c) Starts: *corridor* Target: *cerealbox* in *kitchen*
  The robot explores until it finds *office2*. Upon entry the robot categorises *office2* as kitchen but after further exploration, *office2* is categorised correctly. The robot switches back to exploration and since the kitchen door is closed, it passes kitchen and finds *office1*. Not satisfied with *office1*, the robot gives up since all possible plans success probability are smaller than a given threshold value.
- Fig. 3(d) Starts: *office1* Target:*stapler* in *office2*
  After failing to find the object in *office1* the robot notices the open door, but finding that it is kitchen-like decides not to search the kitchen room. This time the *stapler* object is found in *office2*
- Fig. 3(e) Starts: *kitchen* Target: *cerealbox* in *kitchen*
  As before it tries locating a table, but in this case all table objects have been eliminated beforehand; failing to detect a table the robot switches to looking for a counter. Finding no counter either, it finally goes out in the corridor to look for another kitchen and upon failing that, gives up.
- Fig. 3(f) Starts: *corridor* Target: *whiteboardmarker* in *office1*
  The robot is started in the corridor and driven to the kitchen by a joystick; thus in this case the environment is largely explored already when the planner is activated and asked to find a *whiteboardmarker* object. The part of the corridor leading to *office2* has been blocked. The robot immediately finds its way to *office1* and launches a search which results in a successful detection of the target object.

In the following, we describe the planning decisions in more detail for a run similar to the one described in Fig. 3(a), with the main difference being that the cereals could not be found in the end due to a false negative detection.

The first plan, with the robot starting out in the middle of the corridor, looks as follows:

ASSUME-LEADS-TO-ROOM place1 kitchen
ASSUME-OBJECT-EXISTS table IN new-room1 kitchen
ASSUME-OBJECT-EXISTS cerealbox ON new-object1 table kitchen
MOVE place1
CREATEVIEWCONES table IN new-room1
SEARCHFOROBJECT table IN new-room1 new-object1
CREATEVIEWCONES cerealbox ON new-object1
SEARCHFOROBJECT cerealbox ON new-object1 new-object2
REPORTPOSITION new-object2

Here we see several virtual objects being introduced: The first action assumes that *place1* leads to a new room *new-room1* with category kitchen. The next two assumptions hypothesize that a table exists in the room and that cornflakes exist on that table. The rest of the plan is rather straightforward: create view cones and search for the table, then create view cones and search for the cereal box.

Execution of that plan leads to frequent replanning, as the first assumption is usually too optimistic: most placeholders do not directly lead to a new room, but require a bit more exploration.

After following the corridor, the robot does find the office, and returns to the corridor to explore into the other direction. It finally finds a room which has a high likelihood of being a kitchen.

ASSUME-CATEGORY room3 kitchen
ASSUME-OBJECT-EXISTS table IN room3 kitchen
ASSUME-OBJECT-EXISTS cerealbox ON new-object1 table kitchen
MOVE place17
MOVE place18
MOVE place16
CREATEVIEWCONES table IN room3
SEARCHFOROBJECT table IN room3 new-object1
CREATEVIEWCONES cerealbox ON new-object1
SEARCHFOROBJECT cerealbox ON new-object1 new-object2

The new plan looks similar to the first one, except that we do not assume the existence of a new room but the

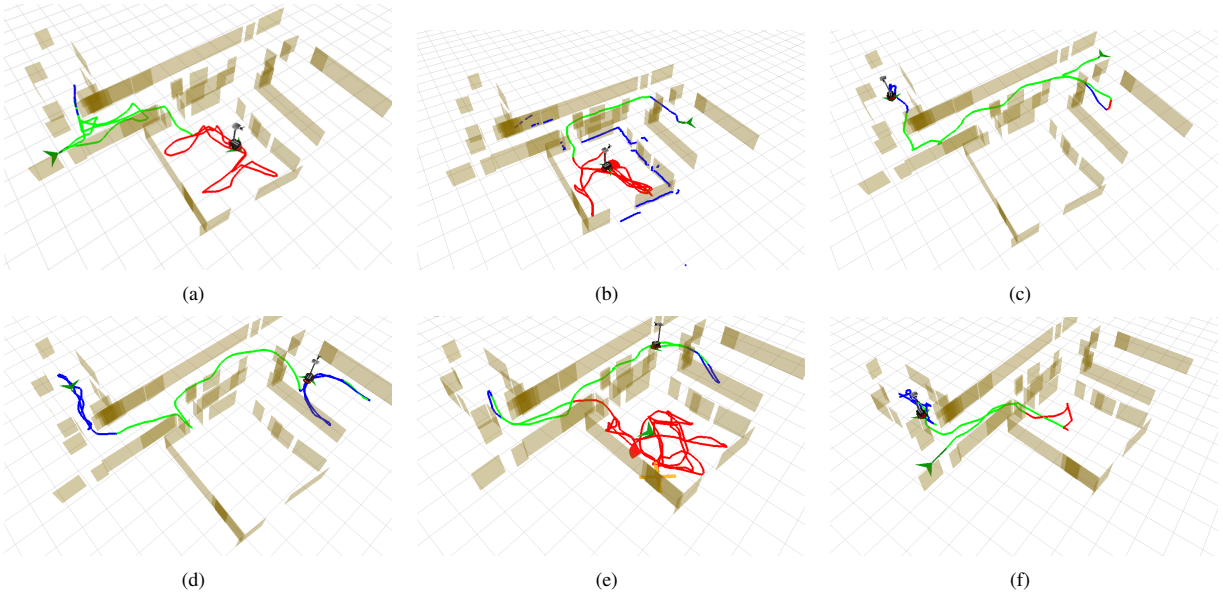


Fig. 3. Trajectories taken by the robot in multiple experiments

category of an existing one. Also, the robot cannot start creating view cones immediately because a precondition of the CREATEVIEWCONES action is that the room must be fully explored, which involves exploring all remaining placeholders in the room.

After view cones are created, the decision theoretic planner is invoked. We used a relatively simple sensing model, with a false negative probability of 0.2 and a false positive probability of 0.05 – these are educated guesses, though. The DT planner starts moving around and processing view cones until it eventually detects a table and returns to the continual planner. At this point the probability of the room being a kitchen is so high, that it considered to be certain by the planner. With lots of the initial uncertainty removed, the final plan is straightforward:

ASSUME-OBJECT-EXISTS cerealbox ON object1 table kitchen
CREATEVIEWCONES cerealbox ON object1
SEARCHFOROBJECT cerealbox ON object1 new-object2
REPORTPOSITION new-object2

During the run, the continual planner created 14 plans in total, taking 0.2 – 0.5 seconds per plan on average. The DT planner was called twice, and took about 0.5 – 2 seconds per action it executed.

V. CONCLUSIONS AND FUTURE WORK

In this paper we have presented a planning approach to the active object search. We made use of a switching planner, combing a classical continual planner with a decision theoretic planner. We provide a model for the planning domain appropriate for the planner and show by experimental results that the system is able to search for objects in a real world office environment making use of both low level sensor perceipt and high level conceptual and semantic information. Future work includes incorporating 3D shape cues to guide the search and a specialized planner for the AVS problem.

REFERENCES

[1] Alper Aydemir, Kristoffer Sjöö, John Folkesson, and Patric 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.

[2] C. Bäckström and B. Nebel. Complexity results for SAS^+ planning. *Comp. Intell.*, 11(4):625–655, 1995.

[3] Michael Brenner and Bernhard Nebel. Continual planning and acting in dynamic multiagent environments. *Journal of Autonomous Agents and Multiagent Systems*, 19(3):297–331, 2009.

[4] Thomas D. Garvey. Perceptual strategies for purposive vision. Technical Report 117, AI Center, SRI International, 333 Ravenswood Ave., Menlo Park, CA 94025, Sep 1976.

[5] Moritz Göbelbecker, Charles Gretton, and Richard Dearden. A switching planner for combined task and observation planning. In *Twenty-Fifth Conference on Artificial Intelligence (AAAI-11)*, August 2011.

[6] Marc Hanheide, Charles Gretton, Richard W Dearden, Nick A Hawes, Jeremy L Wyatt, Andrzej Pronobis, Alper Aydemir, Moritz Göbelbecker, and Hendrik Zender. Exploiting Probabilistic Knowledge under Uncertain Sensing for Efficient Robot Behaviour. In *Proc. Int. Joint Conf. on Artificial Intelligence (IJCAI)*, 2011.

[7] Malte Helmert. The fast downward planning system. *Journal of Artificial Intelligence Research*, 26:191–246, 2006.

[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. 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.

[10] T. Mörwald, J. Prankl, A. Richtsfeld, M. Zillich, and M. Vincze. BLORT - The blocks world robotic vision toolbox. In *Workshop on Best Practice in 3D Perception and Modeling for Mobile Manipulation at ICRA 2010*, 2010.

[11] Andrzej Pronobis and Patric Jensfelt. Hierarchical multi-modal place categorization. In *submitted to ECMR'11*, 2011.

[12] Andrzej Pronobis, Oscar M. Mozos, Barbara Caputo, and Patric Jensfelt. Multi-modal semantic place classification. *The International Journal of Robotics Research (IJRR), Special Issue on Robotic Vision*, 29(2-3):298–320, February 2010.

[13] J. K. Tsotsos. On the relative complexity of active vs. passive visual search. *International Journal of Computer Vision*, 7(2):127–141, 1992.

[14] 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.

[15] Yiming Ye and John K. Tsotsos. Sensor planning for 3d object search. *Comput. Vis. Image Underst.*, 73(2):145–168, 1999.

Large-scale Semantic Mapping and Reasoning with Heterogeneous Modalities

Andrzej Pronobis and Patric Jensfelt
{pronobis, patric}@kth.se

***Abstract*— This paper presents a probabilistic framework combining heterogeneous, uncertain, information such as object observations, shape, size, appearance of rooms and human input for semantic mapping. It abstracts multi-modal sensory information and integrates it with conceptual common-sense knowledge in a fully probabilistic fashion. It relies on the concept of spatial properties which make the semantic map more descriptive, and the system more scalable and better adapted for human interaction. A probabilistic graphical model, a chain-graph, is used to represent the conceptual information and perform spatial reasoning. Experimental results from online system tests in a large unstructured office environment highlight the system's ability to infer semantic room categories, predict existence of objects and values of other spatial properties as well as reason about unexplored space.**

I. INTRODUCTION

In this paper we deal with the problem of modeling space in order to understand it, reason about it and be able to act efficiently in it. We consider applications where the robot is operating in indoor office or domestic environments, i.e. environments which have been made for and are, up until now, almost exclusively inhabited by humans. In such an environment human concepts such as rooms, objects and properties such as the size and shape of rooms are important, not only because of the interaction with humans but also for generating efficient robot behavior, knowledge representation and abstraction of spatial knowledge. This is what we mean by semantic mapping. The semantic mapping system we present will be used in the context of a mobile robot (see Fig. 1) but most of the system would remain unchanged if for example used as part of a wearable device.

This paper builds on our previous work [7], [16] and now focuses on semantic mapping presenting a complete semantic mapping system with several contributions also at a component level. The system makes use of multi-modal sensory information, including information gathered from humans where humans are attributed a "sensor model" just like other sensors. It supports inference about unexplored concepts (e.g. objects, rooms) and allows for goal oriented exploration using a distribution of possible extensions to the known world. We present an extensive experimental evaluation, both offline and online where the whole system runs in real-time on an entire office floor.

A unique feature of our system is the ability to extract semantic information from multiple heterogeneous modalities and integrate it in a principled manner with conceptual common-sense knowledge in a fully probabilistic fashion. The system combines information about the existence of objects, landmarks, the appearance, geometry and topology of space as well as human asserted input. This is possible thanks to an architecture based on semantic properties of spatial entities. The properties correspond to human concepts of space and permit creation of a more descriptive spatial representation in which all entities have attributes as shown in Fig. 1 (e.g. large, square double office with multiple books).

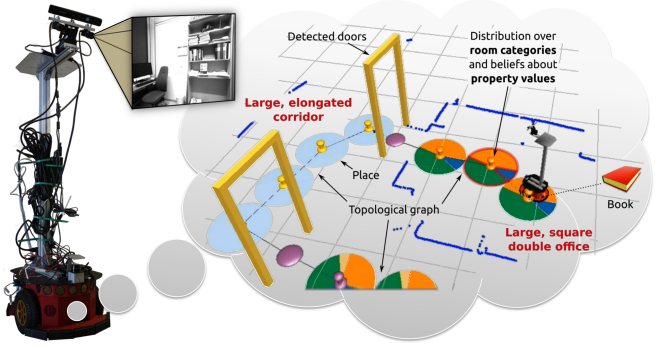


Fig. 1. Our robot platform and an illustration of a semantic map.

The presented approach is evaluated offline on a new comprehensive database, COLD-Stockholm, capturing appearance and geometry of almost 50 rooms belonging to different semantic categories as well as online in the same environment on a mobile robot. A video illustrating the system in action is available online at:

http://www.semantic-maps.org

The remaining sections first relate this work to other approaches in the literature and then discuss the problem of spatial understanding and present our framework from the representational and systems point of view. This is followed by a detailed presentation of our conceptual mapping and reasoning component and experimental evaluation.

II. RELATED WORK

The semantic mapping problem has only recently received significant attention. There exists a broad literature on mobile robot localization, mapping, navigation and place classification [3], [4], [20], [23], [19], [17]. Every such algorithm maintains a representation of space and performs spatial reasoning. However, this representation is usually specific to the particular problem and only captures a fraction of the broad spectrum of spatial knowledge. Other, more general frameworks, such as the Spatial Semantic Hierarchy [9] concentrate on lower levels of spatial knowledge abstraction

This work was supported by the SSF through its Centre for Autonomous Systems (CAS) and the EU FP7 project CogX. The help by Alper Aydemir, Moritz Göbelbecker and Kristoffer Sjöö is also gratefully acknowledged.

Authors are with Centre for Autonomous Systems, KTH Royal Institute of Technology, Stockholm, Sweden.

	Place appearance	Place geometry	Object information	Topology	Human input	Segmentation	Conceptual map	Uncertain concepts	Inferring properties	Concepts acquired
[6]			✓			✓	✓		✓	
[25]		✓	✓		✓	✓	✓		✓	
[21]			✓			✓		✓		✓
[11]			✓							
[22]			✓			✓		✓	✓	✓
[15]		✓	✓							
[14]					✓	✓				
This work	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

TABLE I

PROPERTIES OF VARIOUS SEMANTIC MAPPING APPROACHES

and do not support higher-level conceptualization or representation of categorical information.

Table I compares properties of various semantic mapping approaches for indoor environments. None of the listed methods uses topology of the environment or general appearance of places as a source of semantic information. This is surprising given the large body of work on appearance-based place categorization [20], [23], [19], [17]. Two methods, [25] and [15] make use of geometric place information extracted from laser range sensors, and only [25] applies a previously developed place classification technique for this purpose. In [25], semantic cues can be obtained by a situated dialogue with a user and [14] build maps augmented with semantic symbols purely from human input. Almost every method is focused primarily on using objects for extracting spatial semantics [6], [25], [21], [11], [22], [15]. Objects clearly carry a lot of semantic information; however, they are also sparse and reliable object categorization in real-world environments is still a major open challenge. At the same time, valuable semantic cues are also encoded in geometry, general appearance and topology. The inability to fuse together all the sources of information is likely a result of the different character of the different inputs. In this work, we present a system able to combine all the aforementioned sources of semantic information: general appearance and geometry of places, object information, topological structure and human input.

The conceptual map in our system is also a unique feature. The most comprehensive related representations has been proposed in [6] and [25]. Both approaches encode an ontology of an indoor environment. However, those ontologies are built manually and use traditional AI reasoning techniques which are unable to incorporate uncertainty that is inherently connected with semantic information obtained through robot sensors in realistic environments. In contrast, we implement a probabilistic ontology and a probabilistic inference engine incorporating uncertainty in definitions of concepts and their links to instances of spatial entities. Moreover, the values of all properties for which direct evidence is not available can be inferred based on all the available semantic information. Additionally, as in case of [21] and [22] the concept definitions are acquired automatically from online databases and floor plans obtained from robotics datasets. Finally, we have shown [7], [1] that our approach can be combined with general planning components and is suitable for generating active robot behavior in a similar fashion to [22].

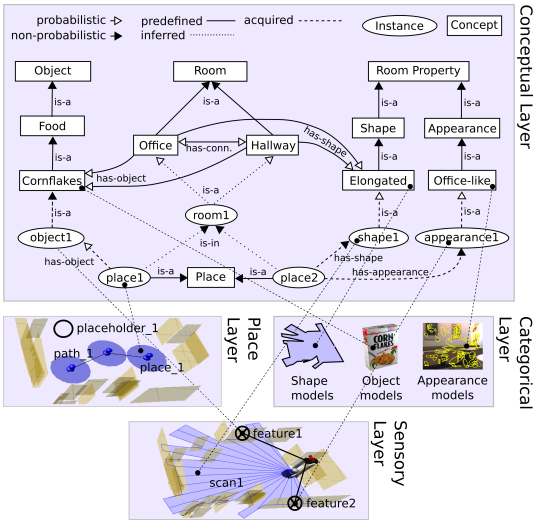


Fig. 2. The layered structure of the spatial representation and a visualization of an excerpt of the ontology of the conceptual layer. The conceptual layer comprises knowledge about concepts (rectangles), relations between those concepts and instances of spatial entities (ellipses).

III. SEMANTIC SPATIAL UNDERSTANDING

The functionality of our system is centred around the representation of complex, cross-modal, spatial knowledge that is inherently uncertain and dynamic. The representation employed here follows the principles presented in [18].

The primary assumption in our approach is that spatial knowledge should be abstracted to keep the representations compact, make knowledge more robust to dynamic changes, and allow the robot to infer additional knowledge about the environment based on combining background knowledge with observations. As one example of abstraction, we discretize the continuous space into discrete areas called *places*. Places connect to other places by *paths* which are generated as the robot travels between them forming a topological map. Hypothesized places, referred to as *placeholders*, are generated in the unexplored parts of space close to areas visited by the robot. This permits reasoning about unknown space [24]. An important concept employed by humans in order to group locations is a *room*. Rooms tend to share similar functionality and semantics and are typically assigned semantic categorical labels e.g. a double office. This make them appropriate units for knowledge integration over space.

A. Spatial Knowledge Representation

The structure of the spatial knowledge representation is presented in Fig. 2. The framework comprises four layers, each focusing on a different level of knowledge abstraction, from low-level sensory input to high-level conceptual symbols.

The lowest level of our representation is the sensory layer which maintains an accurate representation of the robot's environment corresponding to a metric map in our system. Above, the place layer contains the place, paths and placeholders. The categorical layer comprises universal

categorical models (in our case static). These models describe objects and landmarks, as well as spatial properties such as a geometrical models of room shape or a visual models of appearance. On top is the conceptual layer, which is the primary focus of this paper. It is populated by instances of spatial concepts and creates a unified representation relating sensed instance knowledge from lower-level layers to general common-sense conceptual knowledge. Moreover, it includes a taxonomy of human-compatible spatial concepts. It is the conceptual layer which would contain the information that kitchens commonly contain cereal boxes and have a certain appearance and allows the robot to infer that the cornflakes box in front of the robot makes it more likely that the current room is a kitchen.

B. Conceptual Knowledge Representation

A visualization of the data representation of the conceptual layer is shown in Fig. 2. This representation is *relational*, describing common-sense knowledge as relations between concepts (e.g. kitchen has-object cornflakes), and describing instance knowledge as relations between either instances and concepts (e.g. object1 is-a cornflakes), or instances and other instances (e.g. place1 has-object object1). Relations in the conceptual map are either *predefined*, *acquired*, or *inferred*, and can either be deterministic or probabilistic. Probabilistic relations allow the expression of statistical dependencies and uncertainty as in the case of the "kitchen has-object cornflakes" or "room1 is-a hallway" relations which holds only with a certain probability. An acquired relation is one that is grounded in observations and generated as a result of a perceptual process. Predefined relations are given (and quantified in the case they are probabilistic) as part of a fixed ontology of common-sense knowledge. Inferred relations are the result of inference processes operating solely on the conceptual map.

The representation defines a taxonomy of concepts and associations between instances and concepts using hyponym relationships (is-a). Then, directed relations (has-a) are used to describe properties of room categories in terms of spatial properties, such as shape, size or appearance, and objects. Finally, we use undirected associative relations to represent connectivity between rooms.

IV. SEMANTIC MAPPING

A. Property-based Semantic Mapping

An important paradigm underpinning the design of our semantic mapping approach is the use of *properties of space*. Properties can be seen as attributes characterizing discrete spatial entities identified by the robot, such as places or placeholders. Additionally, properties can correspond to human concepts and thus provide another layer of spatial semantics shared between the robot and the user. The values of properties can be inferred from observations and other properties. Properties result from interpreting specific sensory information directly. They are modality specific and each property is connected to a model of sensory information. Higher level concepts, such as room categories, are defined based on the properties. As a result, to the conceptual reasoning, properties serve as connections between higher level concepts and low-level observations. Moreover, they permit building more specialized concepts that would be difficult to infer from uni-modal observations. The idea of using an intermediate level of properties in a feed-forward manner for place categorization has been evaluated previously as a proof of concept [16]. In this work, we generalize beyond a pure feed-forward strategy, so that both properties and room categories influence each other and provide a much more complete representation of space. Hence, we can define the problem of semantic mapping as that of estimating the joint probability distribution over categorical room labels and all values of properties of space for all places.

The current implementation of our system utilizes several types of properties assigned to places:

- *objects* - each object class results in one property associated with a place encoding the expected/observed number of such objects at a certain place
- *doorway* - determines if a place is located in a doorway
- *shape* - geometrical shape of a place extracted from laser data (e.g. elongated, square)
- *size* - size of a place extracted from laser data (e.g. large (compared to other typical rooms))
- *appearance* (e.g. office-like appearance) - visual appearance of a place

In addition to the properties of places, placeholders also have:

- *associated space* - the amount of visible free space around the placeholder not yet assigned to any place

For details about estimation of the placeholder property values, see [24]. We maintain a probability distribution over the property values in the system.

The property-based architecture has several advantages. First, it provides fine-grained and more descriptive representation of space. This can enhance the quality of human-robot interaction, increasing the robot's ability to understand referring expressions and acquire spatial knowledge directly from humans as well as human's understanding of the robot's internal spatial knowledge. The additional semantic knowledge can also be used for generating a more efficient robot's behavior, for example on the task of finding objects in large-scale environments [7], [1].

The approach has many of the advantages of high-level sensor fusion which was shown to outperform low-level feature integration for several problems (see [17] and references therein). It allows for integration of heterogeneous modalities and various types of models adapted to the characteristics of each modality (e.g. robust kernel-based discriminative models for high-dimensional data and probabilistic generative models for data of lower dimensionality or conceptual knowledge). Finally, it enhances the scalability of the approach 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 models. This saves memory (models of visual data can be hundreds of megabytes in size) and saves computations (calculations shared across categories).

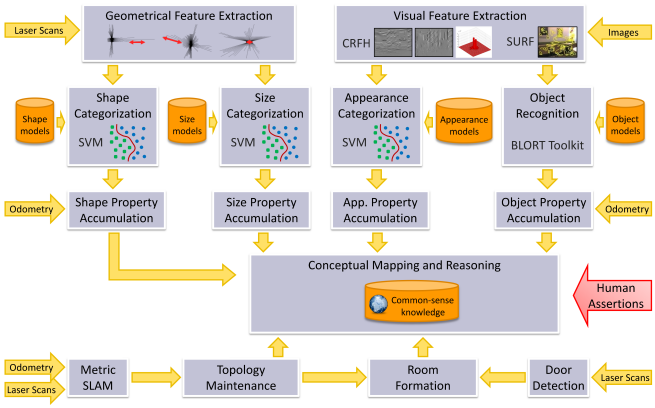


Fig. 3. Structure of the system and data flow between its main components.

The introduction of properties also facilitates training. Once models associated with properties are trained, training the system for a new category is decoupled from low-level sensor data.

B. The Semantic Mapping System

A visualization of the system components and data flow is presented in Fig. 3 and follows the principles outlined above. The layered structure of the spatial knowledge representation as well as the property-based architecture naturally permit the existence of data driven processes that abstract and integrate knowledge. In order to make those processes tractable, the updates of more abstract representations is performed only if a discrete value changes or a continuous values changes above a certain threshold (selected manually).

First, mapping and topology maintenance processes create the topological graph of places, paths and placeholders. A SLAM algorithm [5] builds a metric map of the environment. In our implementation the places are spread out over space like bread crumbs every one meter [25]. Unexplored space is covered with placeholders indicating location of potential places that can be discovered through exploration [24]. This approach to space discretization is limited and requires maintaining a global metric map of the environment. Vision-based topological mapping algorithms such as [4] could be used instead.

In the case of indoor environments, rooms are usually separated by doors or other narrow openings. Thus, we currently use the doorway place property in order to form rooms. A simple, template-based door detector [8] operates on laser range data and the doorway property of a place is set depending on whether the place is located inside a doorway. Then, based on the information about the connectivity of places and the doorway property value, a process forms rooms by clustering places that are transitively interconnected without passing a doorway. Since the door detection algorithm can produce false positives and false negatives, room formation is using non-monotonic inference as described in [25]. We intend to involve all properties of space for room segmentation in the future.

The categorical sensory models of properties are continuously classifying the robot's observations obtained from the laser range finder and a camera. The estimated classification confidence information for each property value is then accumulated over each of the viewpoints observed by the robot while being in a certain place using a spatio-temporal accumulation algorithm presented in [17] and further normalized to form probabilities. The outcomes are then compared to previous observations in order to detect significant changes and fed into the conceptual mapping and reasoning component where they trigger probabilistic inference. If available, human asserted knowledge is provided to the conceptual mapping component where it is combined with the property values.

The resulting system operates in real-time on a standard laptop and is capable of semantic mapping of large scale environments. Since the probabilistic conceptual inference is computationally very efficient, it requires only a small fraction of the computational power. The system scales well not only with the number of room categories, but also with the size of the environment. The system dynamically segments space and integrates knowledge over time, space and multiple information sources. The next sections provide details about the sensory models as well as the the conceptual mapping component.

V. SENSORY MODELS OF PROPERTIES

To extract the semantic properties of spatial entities, the system employs a set of categorical models of sensory information. These models are implemented according to established object and scene modeling approaches.

a) Geometrical Property Models: Two independent models of shape and size properties are built. In both cases we use a set of simple geometrical features extracted from laser scans, as proposed in [17]. To provide sufficient robustness and tractability in the presence of noisy, high-dimensional information, we use kernel-based discriminative classifiers, namely Support Vector Machines (SVM) (see [17] for details). The models are trained from sequences of laser scans recorded in multiple instances of rooms of different shape and size. By including several different room instances into training, the acquired model can generalize sufficiently to provide categorization rather than instance recognition. We identified 3 room shapes (elongated, rectangular and square) as well as 3 room sizes (small, medium and large).

b) Appearance Property Models: We built two different models of general visual appearances of places, one for global and one for local image representation. The former was built from the Composed Receptive Field Histograms (CRFH) [17] calculated over the whole image, while the latter from local SURF features quantized into visual words [2]. The outputs of the two models were further integrated using the Generalized Discriminative Accumulation Scheme (G-DAS [17]). The models were trained on image sequences acquired in rooms belonging to various categories under different illumination conditions in order to generalize to new environments. We identified 7 different appearances: anteroom-like, bathroom-like, hallway-like, kitchen-like, lab-like, meetingroom-like, office-like.

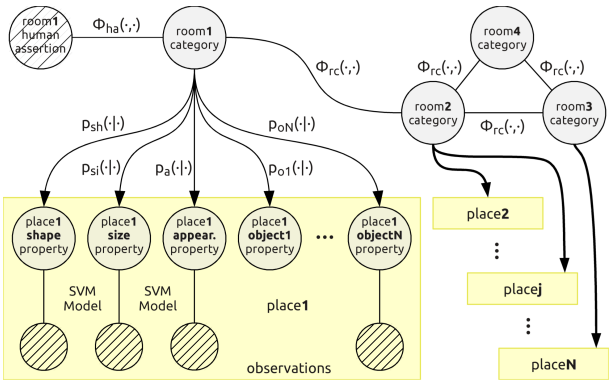


Fig. 4. Structure of the chain graph model of the conceptual map. The vertices represent random variables. The edges represent the directed and undirected probabilistic relationships between the random variables. The textured vertices indicate observations that correspond to sensed evidence.

c) Object Models: To model objects, we used the approach taken from the BLORT toolkit [13] based on SIFT recognition. We trained 6 object instance models for objects belonging to categories typically find in office environments: a book, a cereal box, a computer, a robot, a stapler, and a roll of toilet paper.

VI. PROBABILISTIC CONCEPTUAL MAPPING AND REASONING

To fully exploit the uncertainties provided by the sensory models of properties and permit uncertain spatial reasoning, the conceptual map is represented as a probabilistic *chain graph model* [10]. The structure is adapted at runtime according to the state of the underlying topological map. This is a unique feature of our approach compared to other semantic mapping systems (see Section II).

Chain graphs are a natural generalization of directed (Bayesian Networks) and undirected (Markov Random Fields) graphical models. As such, they allow for modeling both "directed" causal as well as "undirected" symmetric or associative relationships, including circular dependencies originating from possible loops in the topological graph. In order to perform inference on the chain graph, we first convert it into a factor graph representation and apply an approximate inference engine, namely Loopy Belief Propagation [12], to comply with time constraints imposed by the robotic applications.

A. Conceptual Map

The structure of the chain graph for the conceptual map is presented in Figure 4. Each discrete place instance is represented by a set of random variables, one for each property linked to that place. These are each connected to a random variable for the room category, representing the "is-a" relation between rooms and their categories in Figure 2. Moreover, the room category variables are connected by undirected links to one another according to the topological map. The doorway places are seen as transition areas between rooms and are not represented in the conceptual map. The potential functions $\phi_{rc}(\cdot,\cdot)$ describe knowledge about typical connectivity of rooms of certain categories (e.g. that kitchens are more likely to be connected to corridors than to other kitchens).

The remaining variables represent shape, size and appearance properties of space and the presence of objects. These are connected to observations of features extracted directly from the sensory input. These links are quantified by the categorical models of sensory information. Finally, the distributions $p_{sh}(\cdot|\cdot)$, $p_{si}(\cdot|\cdot)$, $p_a(\cdot|\cdot)$, $p_{o_i}(\cdot|\cdot)$ represent the common sense knowledge about shape, size, appearance, and object co-occurrence, respectively. It is assumed that the same object is never represented twice in the conceptual map and data association between object observations is performed while maintaining the sensory layer.

If human asserted input about room categories or other properties of the system is available, it can be seamlessly integrated with the other sources of information. Human assertions about semantic room categories are included by adding a new variable representing an observation of the human assertion and a potential $\phi_{ha}(\cdot,\cdot)$ representing the relation between the assertion and the room category. Identical procedure can be applied if the asserted knowledge is available about some other property of space, e.g. presence of an object.

B. Representing and Quantifying Relations

In our system, the "has-a" relations for room connectivity, shapes, sizes and appearances represented by the potential $\phi_{rc}(\cdot,\cdot)$ and distributions $p_{sh}(\cdot|\cdot)$, $p_{si}(\cdot|\cdot)$, $p_a(\cdot|\cdot)$, $p_{o_i}(\cdot|\cdot)$ were acquired by analyzing annotations in the database used in this paper. Co-occurrences between room categories of neighboring rooms as well as room categories and property values were counted and later normalized to form distributions.

The conditional probability distributions $p_{o_i}(\cdot|\cdot)$ are represented by Poisson distributions. The Poisson distribution was selected in order to easily model the expected number of object occurrences through its parameter λ as well as the ability to estimate λ from the probability of object existence obtained from common-sense knowledge databases. The probability of existence of an object of a certain category in a certain type of room was first bootstrapped using a part of the *Open Mind Indoor Common Sense* database[1]. Obtained object-location pairs were then used to generate '*obj* in the *loc*' queries to an online image search engine. The number of returned hits were used to obtain the probability value. More details about this approach can be found in [7]

The relations between human assertions and concepts (e.g. $\phi_{ha}(\cdot,\cdot)$) can be used to represent the uncertainty in perception of the human statements as well as a dependency between various assertions and concept values (e.g. both "kitchenette" and "kitchen" might be used to refer to a kitchen). In our system, we assign the potential value 0.8 when the assertion exactly matches the room category and we distribute the potential 0.2 evenly across all the remaining assertions.

[1] `http://openmind.hri-us.com/`

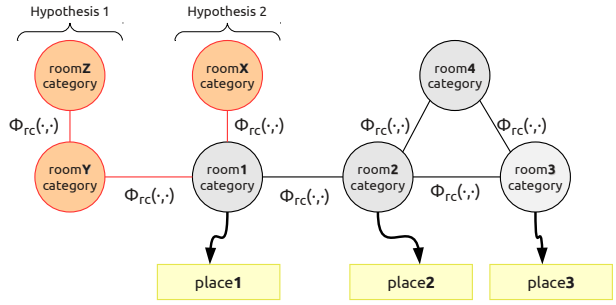


Fig. 5. Examples of extensions of the conceptual map permitting reasoning about unexplored space behind placeholder located in room 1.

C. Reasoning about Unexplored Space

The primary benefits of having a probabilistic relational conceptual representation is its capability to perform uncertain inference about some concepts based solely on their relations to other concepts rather than direct observations. This permits spatial reasoning about unexplored space and we will show two examples of that.

Consider the case of predicting the presence of objects of certain categories in a room with a known category. This can be easily performed in our model by adding variables and relations for object categories without providing the actual object observations. We will show through the experiments that the system is able to continuously predict the existence of objects based on other semantic cues.

Another way of using the predictive power of the conceptual map is to predict the existence of a room of a certain category in the unexplored space behind a placeholder. In such case, the conceptual map is extended from the room in which the placeholder exists with variables representing categories of hypothesized rooms for different possible room configurations in the unexplored space. For each configuration, the categories of the hypothesized rooms are calculated and the obtained probabilities of existence of rooms of certain categories are summed over all possible configurations.

In a simple case, we can consider only three hypotheses: (1) placeholder does not lead to a new room; (2) placeholder leads to a single new room; (3) placeholder leads to a new room connected to another new room. If we assign equal likelihood to the case (2) and (3), it is sufficient to calculate a probability of the placeholder leading to at least one room ($p(r)$). This can be estimated as follows: $p(r) = p(ph)(1 - p(d)) + p(d)$, where $p(ph)$ denotes the probability that the placeholder leads to another placeholder and thus potentially to another room and $p(d)$ is the probability associated with the placeholder doorway property. $p(ph)$ can be estimated from the associated space placeholder property.

VII. EXPERIMENTAL SCENARIO

All the categorical models used in the experiments were trained on the COLD-Stockholm database[2]. Several parts of the database were previously used during the RobotVision@ImageCLEF[3] contests and proved to be challenging in the context of room categorization. The database consists of multiple sequences of image, laser range and odometry data. The acquisition was performed on four different floors (4th to 7th) of an office environment, consisting of 47 areas (usually corresponding to separate rooms) belonging to 15 different semantic and functional categories and under several different illumination settings (cloudy weather, sunny weather and at night). Each data sample is labeled as belonging to one of the areas according to the position of the robot during acquisition. More detailed information about the database can be found online[2].

A. Experimental Setup

In order to guarantee that the system will never be tested in the same environment in which it was trained, we have divided the COLD-Stockholm 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 were used for testing during our offline experiments and the online experiment was performed on the same floor.

For the purpose of the experiments, we have extended the annotation of the COLD-Stockholm database to include the 3 room shapes, 3 room sizes as well as 7 general appearances. The room size and shape, were decided based on the length ratio (elongated $(0, 0.4]$, rectangular $(0.4, 0.8)$, square $[0.8, 1]$) and maximum length of edges (small $[0m, 3m)$, medium $[3m, 5m)$, large $[5m, \infty)$) of a rectangle fitted to the room outline. These properties together with 6 object types defined 11 room categories used in our experiments: an anteroom, a bathroom, a computer lab, a robot lab, a conference hall, a hallway, a kitchen, a meeting room, and three types of offices, a double office, a single office and a professor's office. The three types of offices, the two types of labs as well as the meeting room and conference hall shared appearance properties (office-like, lab-like and meeting room-like respectively) and could only be discriminated by a using a combination of properties.

VIII. EXPERIMENTS

We first build and evaluate the performance of each of the sensory models of properties offline. To build the models, the rooms having the same values of properties were grouped to form the training and validation datasets. Then, parameters of the models were obtained by cross-validation. Finally, all training and validation data were collected together and used to train the final models. The evaluation was performed on test data acquired in previously unseen rooms.

The classification rates obtained for each of the properties and cues are presented in Tab. II. The rates represent the percentage of correct classifications obtained separately for each of the classes, and then averaged in order to exclude the influence of unbalanced testing set. We can see that all classifiers provided a recognition rate above 80%. Additionally,

[2] http://www.semantic-maps.org/db

[3] http://www.robotvision.info

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 OBTAINED FOR EACH PROPERTY AND CUE.

we see that integrating two visual cues (CRFH and BOW-SURF) increased the classification rate of the appearance property by almost 5 percentage points. For an additional analysis of results, we refer the reader to [16].

The obtained models were used in the semantic mapping system during the online experiments. The experiments were performed on the 6th floor of the building where the COLD-Stockholm database was acquired, i.e. in the part which was not used for training. The robot was manually driven through two parts of the environment consisting of 13 different rooms. It performed real-time semantic mapping without relying on any previous observations of the environment. The obtained maps of the two parts of the environment (A and B) as well as the robot trajectory are presented in Fig. 6.

The robot gathered observations of shapes, sizes, appearances and objects present in the environment and performed reasoning about the values of properties and room categories. If an observation of an object of a certain category was not available, the robot reasoned about its existence based on other available information. The robot recorded beliefs about the shapes, sizes, appearances, objects found and the room categories for every significant change event in the conceptual map. The results for the two parts of the environment are presented in Fig. 7. Each column in the plot corresponds to a single event, and the cells show the probabilities assigned to beliefs. For better analysis, compare the results in Fig. 6 and Fig. 7 using the room numbers as a reference.

By analyzing the events and beliefs for part A, we see that the system correctly identified the first two rooms as a hallway and a single office using purely shape, size and general appearance (there are no object related events for those rooms). The next room was properly classified as a double office, and that belief was further enhanced by the presence of two objects of the category "computer". The next room was initially identified as a double office until the robot was given a human assertion that there is a single computer in this room. This was an indication that the room is a single person office that due to its dimensions is likely to belong to a professor. The remaining rooms were correctly identified as single offices (rooms 4 and 5) and a meeting room (room 6).

Looking at part B, we see that the system identified most of the room categories correctly with the exception of a single office (room 2), which due to a misclassification of size was incorrectly recognized as a double office. The robot was first driven to the robot lab, which was correctly categorized thanks to a combination of a appearance information (lab-like) and an object observation (a robot). Remaining rooms were mapped primarily based on general appearance information as well as geometric properties.

In several rooms, we did not provide any object observations (rooms 0, 1 in part A and 0, 2, 3, 5 in part B). Therefore all the object presence beliefs shown in Fig. 7 obtained for those rooms are predictions of unexplored concepts. The experiment showed that the system can deliver good performance by integrating multiple sources of semantic information. As previously mentioned, a video showcasing the system is available online.

IX. CONCLUSIONS AND FUTURE WORK

In this paper we have presented a probabilistic framework combining heterogeneous, uncertain, information such as object observations, the shape, size, appearance of rooms and human input for semantic mapping. A graphical model, more specifically a chain-graph, is used to represent the semantic information and perform inference over it. We used the concept of spatial properties which allowed us to make the knowledge representation more descriptive and pave the way for better scalability. Finally, we showed how to use the representation in order to reason about unexplored concepts.

There are several ways in which the work presented in this report can be extended, however three are of particular importance. First, we intend to look at ways to make the segmentation of space part of the estimation process as is made in PLISS [19], and while doing so, rely on all available properties. Second, we plan to replace the current space discretization approach with a feature-based clustering technique such as in [4]. Finally, we will investigate the problem of detection and learning of novel properties and room categories to pave the way towards fully self-extendable semantic mapping.

REFERENCES

[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," in *Proc. of ECMR'11*.

[2] H. Bay, A. Ess, T. Tuytelaars, and L. J. Van Gool, "Speeded-up robust features (SURF)," *CVIU*, vol. 110, no. 3, 2008.

[3] M. Cummins and P. M. Newman, "Highly scalable appearance-only SLAM - FAB-MAP 2.0," in *Proc. of RSS'09*.

[4] F. Dayoub, G. Cielniak, and T. Duckett, "A sparse hybrid map for vision-guided mobile robots," in *Proc. of ECMR'11*.

[5] J. Folkesson, P. Jensfelt, and H. I. Christensen, "The M-space feature representation for SLAM," *IEEE Tr. on Robotics*, vol. 23, no. 5, 2007.

[6] C. Galindo, A. Saffiotti, S. Coradeschi, P. Buschka, J. A. Fernández-Madrigal, and J. González, "Multi-hierarchical semantic maps for mobile robotics," in *Proc. of IROS'05*.

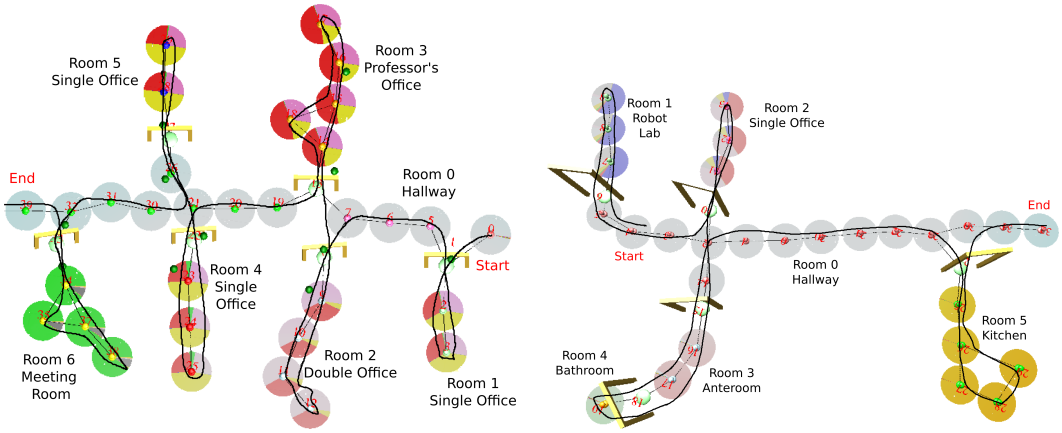
[7] M. Hanheide, C. Gretton, R. W. Dearden, N. A. Hawes, J. L. Wyatt, A. Pronobis, A. Aydemir, M. Göbelbecker, and H. Zender, "Exploiting probabilistic knowledge under uncertain sensing for efficient robot behaviour," in *Proc. of IJCAI'11*, Barcelona, Spain.

[8] P. Jensfelt, "Approaches to mobile robot localization in indoor environments," Ph.D. dissertation, Signal, Sensors and Systems (S3), Royal Institute of Technology, SE-100 44 Stockholm, Sweden, http://www.cas.kth.se/~ patric/publications/phd.html, 2001.

[9] B. Kuipers, "Spatial Semantic Hierarchy," *AI*, vol. 119, no. 1-2, 2000.

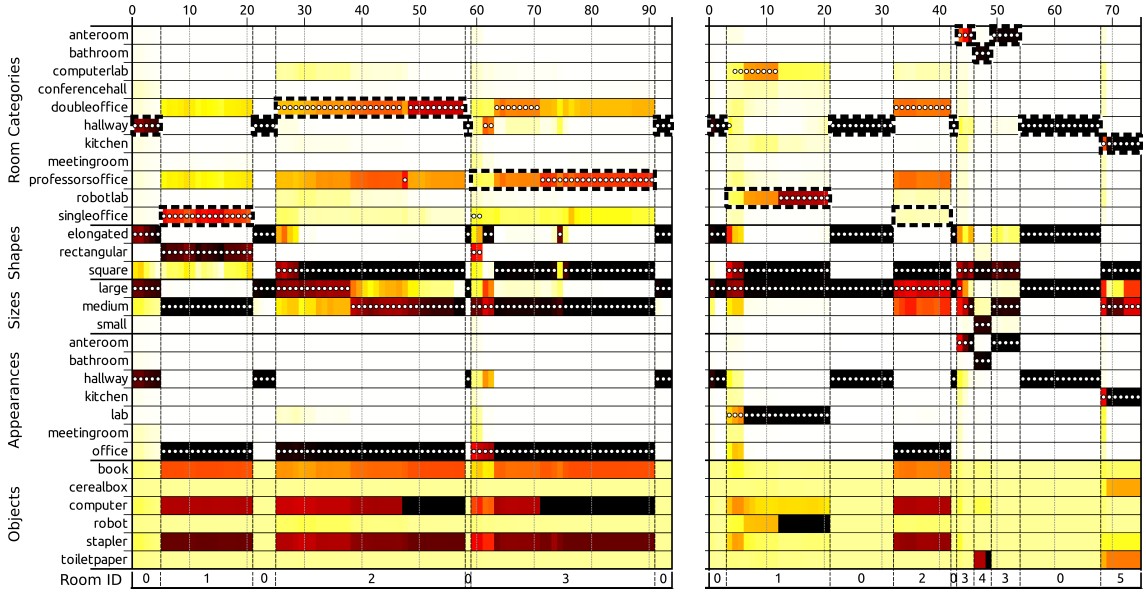
[10] S. Lauritzen and T. Richardson, "Chain graph models and their causal interpretations," *J. of Royal Statistical Society*, vol. 64, no. 3, 2002.

[11] D. Meger, P.-E. Forssen, K. Lai, S. Helmer, S. McCann, T. Southey, M. Baumann, J. J. Little, and D. G. Lowe, "Curious George: An attentive semantic robot," *RAS*, vol. 56, no. 6, 2008.



Part A Part B

Fig. 6. Topological maps of the environment anchored to a metric map indicating the outcomes of room segmentation and categorization. The circles indicate the location of places in the environment and the black line shows the robot's trajectory. The pie charts indicate the probability distributions over the inferred room categories for each room (each fraction of the pie chart corresponds to a room category). In order to see the labels assigned to each fraction as well as a detailed view of the distributions across room categories, objects and values of spatial properties, see Fig. 7.



Part A Part B

Fig. 7. Visualization of the events registered by the system during exploration and its beliefs about the categories of the rooms as well as values of the properties and object presence. Each row represents the development of probabilistic beliefs about a certain concept as the robot explored the environment (see the trajectory in Fig. 6). Darker colors indicate higher probability. The room category ground truth is marked with thick dashed lines. The MAP values are indicated with white dots.

[12] J. M. Mooij, "libDAI: A free and open source C++ library for discrete approximate inference in graphical models," *JMLR*, vol. 11, 2010.

[13] T. Mörwald, J. Prankl, A. Richtsfeld, M. Zillich, and M. Vincze, "BLORT - The blocks world robotic vision toolbox," in *ICRA Workshop Best Practice in 3D Perc. and Model. for Mobile Manipul.*, 2010.

[14] C. Nieto-Granda, J. G. Rogers, A. J. B. Trevor, and H. I. Christensen, "Semantic map partitioning in indoor environments using regional analysis," in *Proc. of IROS'10*, Taipei, Taiwan, 2010.

[15] A. Nüchter and J. Hertzberg, "Towards semantic maps for mobile robots," *RAS*, vol. 56, no. 11, 2008.

[16] A. Pronobis and P. Jensfelt, "Hierarchical multi-modal place categorization," in *Proc. of ECMR'11*.

[17] A. Pronobis, O. M. Mozos, B. Caputo, and P. Jensfelt, "Multi-modal semantic place classification," *IJRR*, vol. 29, no. 2-3, 2010.

[18] A. Pronobis, K. Sjöö, A. Aydemir, A. N. Bishop, and P. Jensfelt, "Representing spatial knowledge in mobile cognitive systems," in *Proc. of IAS'10*.

[19] A. Ranganathan, "PLISS: Detecting and labeling places using online change-point detection," in *Proc. of RSS'10*.

[20] A. Torralba, K. Murphy, W. Freeman, and M. Rubin, "Context-based vision system for place and object recognition," in *Proc. of ICCV'03*.

[21] S. Vasudevan and R. Siegwart, "Bayesian space conceptualization and place classification for semantic maps in mobile robotics," *RAS*, vol. 56, no. 6, 2008.

[22] P. Viswanathan, D. Meger, T. Southey, J. J. Little, and A. K. Mackworth, "Automated spatial-semantic modeling with applications to place labeling and informed search," in *Proc. of CRV'09*.

[23] J. Wu, H. I. Christensen, and J. M. Rehg, "Visual place categorization: problem, dataset, and algorithm," in *Proc. of IROS'09*.

[24] 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*, vol. 2, no. 4, 2010.

[25] H. Zender, O. M. Mozos, P. Jensfelt, G.-J. M. Kruijff, and W. Burgard, "Conceptual spatial representations for indoor mobile robots," *RAS*, vol. 56, no. 6, 2008.

Predicting indoor topology labelings and structure from a large indoor topological database

Alper Aydemir[1] Erik Järleberg[1] Samuel Prentice[2] Patric Jensfelt[1]

[1] CVAP, Royal Institute of Technology, KTH, Sweden
aydemir,erikjar,patric@kth.se
[2] CSAIL, Massachusetts Institute of Technology, USA
prentice@mit.edu

Abstract. A significant amount of research in robotics is aimed towards building robots that operate indoors yet there exists little analysis of how human spaces are organized. In this work we analyze the properties of indoor environments from a large annotated floorplan dataset. We analyze a corpus of 567 floors, 6426 spaces with 91 room types and 8446 connections between rooms corresponding to real places. We present a system that, given a partial graph, predicts the rest of the topology by building a model from this dataset. Our hypothesis is that indoor topologies consists of multiple smaller functional parts. We demonstrate the applicability of our approach with experimental results. We expect that our analysis paves the way for more data driven research on indoor environments.

1 Introduction

Imagine a mobile robot tasked with finding an object on an unexplored office building floor. The robot needs to plan its actions to complete the task of object search and the search performance depends on the accuracy of the robot's expectations. As an example, having found a corridor and an office, its expectation of finding another room by exploring the corridor should be higher than exploring the office as corridors act as connectors in most indoor environments.

In most systems where this type of structural information can be beneficial, the models of indoor environments are hard-coded and not learned from data. Indoor environments are generally organised in interconnected spaces each fulfilling a certain function. A natural way of modeling these environments is by building a graph where each vertex represents a room in the environment and an edge between two vertices indicates a direct, traversable path. Each vertex can have several attributes such as a room category (kitchen, office, restroom etc.), area size and perimeter length. This type of representation is often called a topological map in the literature. More recently, researchers became interested in augmenting topological maps with semantic information by extracting the

aforementioned attributes from sensory data [1, 2, 3]. Although there exists a large body of work on building topological maps, little consideration is given to the analysis and prediction in these maps. One reason for this is building data driven models of topological maps requires collecting data from a high number of actual buildings, recording the floorplan layout including the rooms and adding each room's attributes. This is much harder than an image annotation task.

We leverage on the MIT floorplan database [4], containing 567 floors, 6426 spaces with 91 space categories and 8446 connections between the spaces in total. An example partial topology from the dataset is shown in figure 1. To the best of our knowledge, no previous work exists on the analysis and usage of a dataset of this type and scale. First, we provide an analysis of the topological properties of a large indoor floorplan dataset. Second, we develop a method to predict both the structure (i.e. which type of rooms are connected to each other) and the vertex labelings (i.e. which type of rooms are most commonly found) from a large real-world annotated semantic indoor topology database. We do this on basis of the hypothesis is that indoor environments are topologically arranged in small functional units, e.g. $\{corridor - bathroom - office\}$ or $\{corridor - mailbox - office\}$. Therefore by extracting these frequently occurring topological patterns we can make accurate predictions even though the specific input graph at hand contains rooms of previously unknown categories. Rooms with unknown categories in the input graph corresponds to a real world problem where a robot's classifier may be largely uncertain about a room's category or that the robot has no model for that specific room. Even in this case, the system should still provide reasonable predictions.

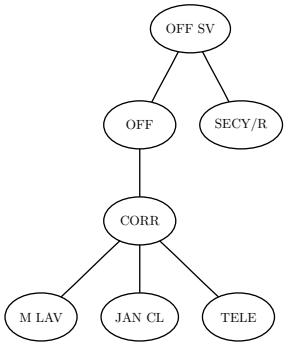


Fig. 1: An example graph from the dataset.

2 Problem Formulation

2.1 Preliminaries

We represent each floor as an undirected graph. Each vertex in a graph is assigned a label from an ordered, finite alphabet such that no two vertices share the same label [5]. A graph is then a three-tuple $G = (V, E, \alpha)$ where V is a finite vertex set, $E \subset V \times V$ is a finite edge set and $\alpha : V \to \mathcal{L}$ is a vertex label mapping. Let $\mathcal{G}$ be the set of all formable graphs using the label alphabet $\mathcal{L}$.

The *graph edit distance* is a notion used to measure how similar two graphs are to each other. It is based upon what are called *edit operations* on a graph. An edit operation is a change performed upon a graph to transform it into a new graph. Normally one considers: vertex substitutions, vertex additions, edge additions, and vertex deletions as possible graph edit operations. We will restrict these operations to two specific types: *edge addition* between two existing vertices in the graph; and, *vertex addition*, which creates a new labeled vertex connected to one of the existing vertices. This is to ensure that we get no disconnected parts and the resulting graphs are connected. With this restriction upon the set of possible edit operations, one cannot always expect to be able to transform an arbitrary graph g_1 into g_2. However if we restrict the domain so that $g_1 \subseteq g_2$ or vice-versa, it is always possible to transform one into the other without considering vertex deletions for example.

We will also denote by $\phi(g_1, g_2)$ the set of possible edit operation sequences transforming g_1 into g_2. Using this we define the distance between two graphs g_1 and g_2 as the minimal cost of transforming one graph into the other: $d(g_1, g_2) = \min_{s \in \phi(g_1, g_2)} c(s)$. It can be shown that this function satisfies the four properties of a metric [6]. We define the ball of a certain radius r to be the set of all graphs which are at most r edit operations away from the graph. That is, $B(G, r) = \{G' \in \mathcal{G} | d(G, G') \leq r\}$.

A *graph database* $\mathcal{D} = \{G_1, ..., G_n\}$ is a set of graphs. Given a graph $G \in \mathcal{G}$ and a graph database $\mathcal{D}$, we define the *projected database* as the set of super-graphs of G. We denote this set as $\mathcal{D}_G = \{G' \in \mathcal{D} | G \subseteq G'\}$. The cardinality of the projected database is called the *frequency* of the graph G in the graph database $\mathcal{D}$ and is denoted by $freq(G) = |\mathcal{D}_G|$.

We may now define the *support* of the graph G as:

$$supp(G) = \frac{freq(G)}{|\mathcal{D}|} \tag{1}$$

A graph G will be called a *frequent subgraph* in $\mathcal{D}$ if $supp(G) \geq \sigma$ where σ is some minimum support threshold, $0 \leq \sigma \leq 1$.

Let $\mathcal{S}$ be the set of frequent subgraphs of the graph database $\mathcal{D}$ for some minimum support threshold σ. That is, $\mathcal{S} = \{G \in \mathcal{D} | supp(G) \geq \sigma\}$

For any given pair of graphs g_1 and g_2, the *Pearson's Correlation Coefficient* describes the linear correlation between the two graphs in the database is defined

as in [7]:

$$\theta(g_1, g_2) = \frac{supp(g_1, g_2) - supp(g_1)supp(g_2)}{\sqrt{supp(g_1)supp(g_2)(1 - supp(g_1))(1 - supp(g_2))}} \tag{2}$$

Finally, the neighbourhood of a vertex v in a graph G will be denoted by $N_G(v)$ or simply $N(v)$ when it is clear which graph is meant. The neighbourhood of v is the induced subgraph of vertices which are adjacent to v in the graph G.

2.2 Formal graph prediction problem formulation

We define the problem as follows. Given a graph database $\mathcal{D}$ we want to find a certain discrete probability distribution. This distribution is an estimate of how probable a certain edit operation upon the current partial graph is. Let $G_p \subset G$ be called the partial graph which is a subgraph of some unknown supergraph G. The *set of all possible next graphs* given a partial graph is the ball of radius one around the partial graph using the graph edit distance metric. That is, the set of all possible next graphs is $B(G_p, 1)$. Once the discrete probability distribution above has been acquired, it is then possible to attain the most probable next graph $G'_p \in B(G_p, 1)$. This graph is simply the result of performing the most probable edit operation upon G_p.

3 The Method

3.1 Analysis of dataset

We start by presenting the insights gained by analyzing the dataset. Each floor in the MIT floorplan dataset consists of a set of *spaces* and their connections to other spaces. Floors can be represented as graphs; the spaces can be interpreted as vertices of a graph and the connections as graph edges [4]. A space can be a room surrounded by walls and accessible via doors, but sometimes a space can also have invisible boundaries, e.g. a coffee shop at the end of a corridor.

Connector spaces such as *corridor* and *stair* are crucial parts of any indoor environment since they act as indoor highways. Our intuition tells us that spaces that have the functionality to connect other rooms and floors together should appear with high frequency in natural indoor environments. Table 1 shows the most frequent vertices in the MIT floorplan dataset with their occurrence frequency in all floors. As can be seen, *corridor* and *stair* are in most floors, ranking as the top two frequent spaces. Offices are also a common space in campus buildings.

Furthermore, we would expect to see some common patterns in floorplans. For example, we would expect certain facilities such as lavatories and elevators to be at easily reachable locations, or connector spaces such as corridors frequently attached to office rooms. Figure 2 shows the most frequent subgraphs in the dataset for graph sizes 3, 4 and 5. It is remarkable that even for large graph sizes with 4 and 5 vertices, certain patterns are commonplace in the dataset. This

supports the hypothesis that indoor topologies consist of commonly occurring smaller parts.

Figure 3a shows the Pearson's correlation coefficient [7] (explained in section 2.1) for the frequent subgraphs in the dataset which occur in more than 16% of all graphs (the frequent subgraph set $\mathcal{S}$ with $\sigma = 0.16$). The graphs are ordered such that the top left pixel is the most frequently occurring subgraph and the top right pixel corresponds to the least frequent. Each pixel represents a frequent subgraph pair and brightness corresponds to high correlations. As an example, figure 3b and 3c correspond to pixel (19,12) or (12,19), which is the highest correlated pair found in this set. Having observed for example the graph in figure 3b, we could say that the edit operation leading to the graph in figure 3c is very probable. The corresponding edit operation would be an *edge addition*, adding an edge between the "OFF" and "P CIRC" vertices.

3.2 Method I

Given an initial input graph G_p, we first compute its projected database $\mathcal{D}_{G_p}$. Then, for each graph $E, E \in \mathcal{D}_{G_p}, E \in B(G_p, 1)$, we calculate the edit operation from G_p to E. Finally, the edit operation whose resulting graph has the highest support is determined. This algorithm is naive in the sense that it considers the whole graph at once. This is akin to a hidden Markov Model formulation where the state of the model is the graph itself and actions are edit operations.

The algorithm performs well for small graph sizes. This is encouraging, however we would expect the naive method to fail for larger graphs. By taking into account the overall structure of G_p (defined in section 2.1) as a whole, the algorithm fails to capture the functional patterns with which humans have designed indoor floorplans. As an example, when a rare vertex is connected to a frequently occuring part of the input graph, the algorithm only considers those graphs which include the rare vertex disregarding others, ignoring the functional aspect of subparts of an indoor topology.

Table 1: Most frequent spaces in the dataset. Here "JAN CL", "ELEC", "OFF SV" are abbreviations for janitor closet, electricity cabinet and office service, respectively.

Vertex	Support
STAIR	85%
CORR	78%
OFF	67%
OFF SV	60%
ELEC	60%
JAN CL	57%
LOBBY	48%

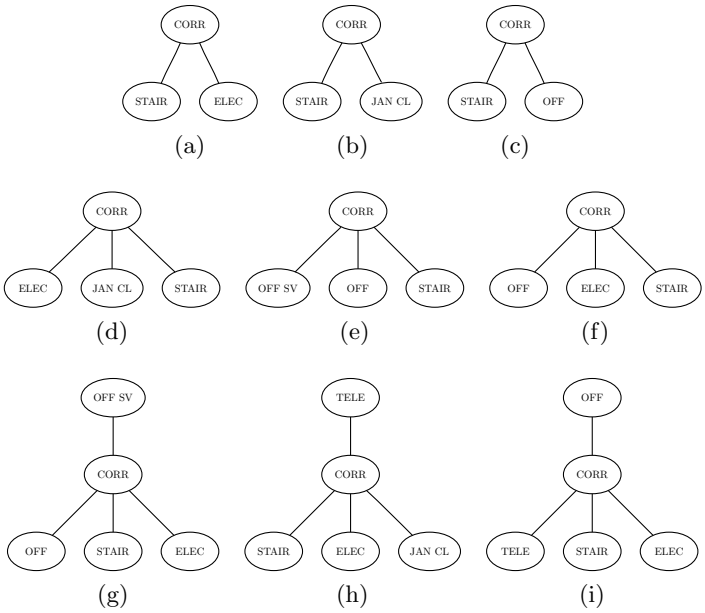


Fig. 2: The three most common frequent subgraphs for graph sizes 3, 4 and 5. The frequencies for subgraphs shown in figures 2a-2c are 37.66, 37.11, 36.56, for figures 2d-2f they are 26.50, 25.04, 25.04 and finally for figures 2g-2i they correspond to 17.18, 17.00, 17.00, respectively.

3.3 Method II - Prediction with Graph Splitting

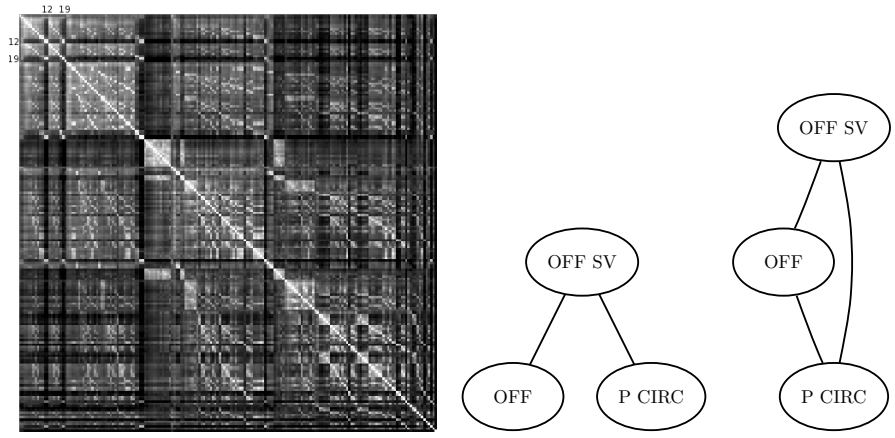
In this method, we make use of the frequently occuring subgraphs in the database. We extract frequent subgraphs using the gSpan Algorithm [8]. This provides us with a frequent subgraph database $\mathcal{S}$ which is used in the first step of the prediction. See figure 4a.

The main steps of this method is given in the following:

1. Split the input partial graph into smaller, overlapping subgraphs which are included in $\mathcal{S}$.
2. For each of these subgraphs of the partial graph, determine the probability of every possible edit operation.
3. Combine the results of the estimates of the edit operations for each subgraph into a final solution for the whole partial graph.

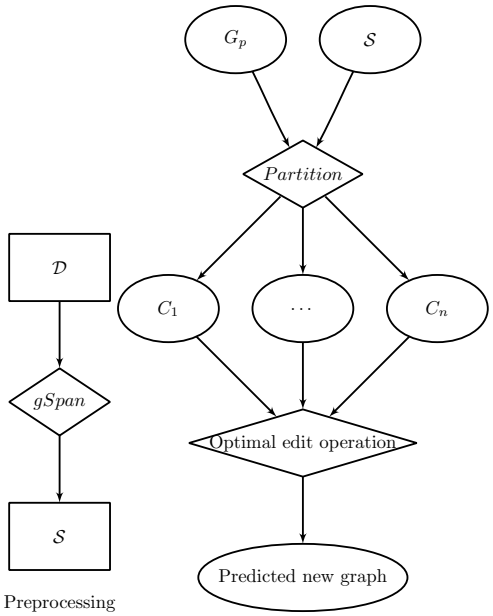
These three steps are summarized in figure 4b.

Step 1: The aim of this step is to divide the partial input graph G_p into a set of overlapping connected subgraphs C where $\forall x \in C, \exists y \in C, x \cap y \neq \emptyset$. The procedure for computing C is given in algorithm 1. The selection of subgraphs



(a) (b) (c)

Fig. 3: 3a: Pearson's correlation coefficient for frequent subgraphs occurring in more than 16% of all graphs in dataset. Each pixel represents a frequent subgraph pair and brightness corresponds to high correlations. Subgraphs are ordered by frequency descending from top left pixel. 3b and 3c show the highest correlated pair, corresponding to pixel (19,12) or (12,19) in 3a.



(a) Preprocessing (b) Prediction

Fig. 4: Predition Algorithm Overview. (a) Frequent subgraphs $\mathcal{S}$ are extracted from the graph dataset $\mathcal{D}$. (b) In each iteration, edit operations are hypothesized on selected subsets of the input graph G_p, and the optimal edit operation is executed.

plays an important role in prediction quality. We pick the elements of C as much as possible from the frequent graph set $\mathcal{S}$. The rationale behind this is that since indoor topologies consists of multiple functional smaller parts, the algorithm should try to identify those and later expand them as viable predictions. First we determine which of the frequent subgraphs from $\mathcal{S}$ that are present in the current partial graphs, and extract the largest possible such frequent subgraphs set and call it P.

In short, algorithm 1 iteratively checks for the elements of S which are included in G_p (the set P) and which share at least one vertex with the list of subgraphs found so far, C, so as to disregard disconnected subgraphs. Another reason is that computing the list of all possible connected subgraphs of G_p becomes intractable even for small-sized graphs. Therefore we utilize the frequent subgraphs of the graph database to bootstrap this computation and cut down the search space.

Step 2: In this step, we aim to calculate the probability of all possible edit operations for each subgraph of G_p. Let $\mathcal{D}_{C_i}$ be the projected database of any subgraph C_i of G_p, that is, the set of all those graphs which are supersets of C_i. Let x be some graph which is one edit operation away from C_i, that is $x \in B(C_i, 1)$. We then define $\phi(x, C_i) = |\{G' \in \mathcal{D}_{C_i} | x \subseteq G'\}|$. That is $\phi(x, C_i)$ gives the number of times we've observed a specific edit operation upon C_i among all the graphs. The most likely edit operation to perform given that we've observed the subgraph C_i is then given by $\underset{x \in B(C_i,1)}{\arg\max}\, \phi(x, C_i)$. This procedure is given in detail in algorithm 3.

Step 3: Given that we have calculated the most likely edit operation for each of the subgraphs $C_1, ..., C_n$, we have for each of these an optimal edit operation leading to new graphs $C'_1, ..., C'_n$ respectively. We must select one of these, and for any selection C'_j made, the resulting prediction will be $G'_p = \bigcup_{i \in [1,n] \setminus \{j\}} C_i \cup C'_j$. We simply select the edit operation which has the highest support from the graph database. That is, $\underset{C_i, i \in [1,n]}{\arg\max} \phi(C_i, C'_i)$.

Given the function $\phi : \mathcal{G} \times \mathcal{G} \to \mathbb{N}$, it is possible to arrive at an estimate of the discrete probability distribution of the different edit operations upon G_p. The distribution is calculated in a frequentist manner and is given by:

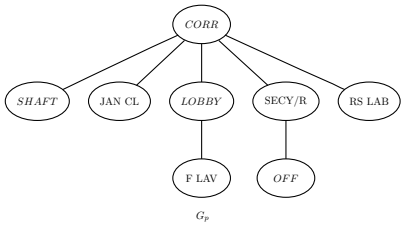
$$P(G'_p = x) = \frac{\phi(x, C_j)}{\sum_{y \in B(C_j,1)} \phi(y, C_j)}, x \in B(C_j, 1) \tag{3}$$

C_j here refers to the selected subgraph and is chosen as detailed above.

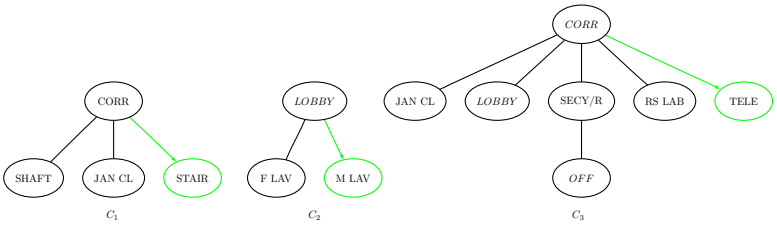
Figure 5a shows the initial partial graph which is the input to the prediction algorithm. In this example the input graph is divided into three subgraphs. The output of the first step of the algorithm is shown in black in figure 5b, 5c and 5d. In the second step, the predicted edit operation with the highest support for each subgraph C_i is shown in green. Finally, in the third step, the edit operation with the highest probability is selected.

This splitting of the input graph agrees with the claim that indoor topologies consist of smaller functional parts. Figure 5b shows that while some vertices may

be rare (such as "SHAFT"), they can occur in a frequent subgraph pattern, in this case forming a "maintenance" functional group. Figure 5d shows a very common structure, with a corridor as a root node. Finally, in figure 5c, we can see that the algorithm has identified a lobby group. This is also quite common, that the lobby acts as a root node similar to a corridor vertex connected in a tree-like structure.



(a) The input partial graph.



(b) First subgraph (c) Second subgraph (d) Third subgraph

Fig. 5: The overlapping subgraphs of a partial graph.

4 Experiments

4.1 Example runs

To illustrate the method, five different states of a prediction sequence are shown in figure 6. The complete unknown graph G is shown in black dashed lines. The starting initial graph is shown in blue. A predicted edit operation existing in G is shown in green and if it does not exist in G, then it is shown in red.

In figure 6a, the partial graph only consists of the female lavatory vertex "F LAV". The prediction algorithm is then applied to produce figure 6b. The next likely edit operation is to add a corridor "CORR" and connect it to the "F LAV". Next in figure 6c, we can see the result of executing the prediction algorithm upon the previous graph consisting of "F LAV" and "CORR". Given that we have observed "F LAV" and "CORR", the algorithm suggests that it is plausible to have a male lavatory "M LAV" connected to the corridor as well.

Algorithm 1 Graph splitting

Input:

- G_p, the current partial graph

Output:

- $C = \{C_1, ..., C_m\}$, the overlapping subgraphs of the partial graph

$P \leftarrow \emptyset$
for $s \in \mathcal{S}$ **do**
  if $s \subseteq G_p \wedge (\neg \exists s' \in \mathcal{S}, s \subseteq s', s' \subseteq G_p)$ **then**
    $P \leftarrow P \cup \{s\}$
  end if
end for
{P now contains those frequent subgraphs which are contained in the partial graph G_p. They are also the largest possible frequent subgraphs. }
sort(P) by graph size, descending.
$C \leftarrow \{\text{FindCommonFreqSubgraph}(P, G_p, \emptyset)\}$
while $|G_p| \neq |\bigcup_{i=1}^{n} C_i|$ **do**
  $Found \leftarrow 0$
  for all $c \in C \wedge Found = 0$ **do**
    $c' \leftarrow \text{FindCommonFreqSubgraph}(P, c, C)$
    if $c' \neq \emptyset$ **then**
      $C \leftarrow C \cup c'$
      $Found \leftarrow 1$
      break
    end if
  end for
  if $Found = 0$ **then**
    $D_g \leftarrow G_p \setminus \bigcup_{i=1}^{n} C_i$
    Add the following vertex set to D_g: $\bigcup_{v \in V(D_g)} N(v, G_p) \setminus D_g$
    Add the edges (from the edge set of G_p) which correspond to the vertex additions above.
    $C \leftarrow C \cup \text{GetComponents}(D_g)$
    return C
  end if
end while
return C

Algorithm 2 FindCommonFreqSubgraph

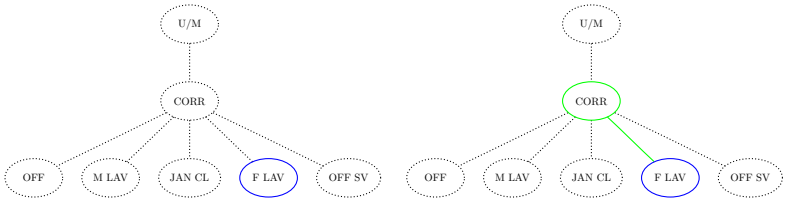
This function will attempt to find another frequent subgraph from the set P that has some vertex in common with some graph C_i (the already established subgraphs of G_p).
Input:

- P, the sorted sequence of frequent subgraphs that are present in the partial graph
- G, a graph which the result should have some vertex in common with, this is always some C_i except for the initial execution.
- $C = \{C_1, ..., C_n\}$, the thus far added overlapping subgraphs of the partial graph

Output:

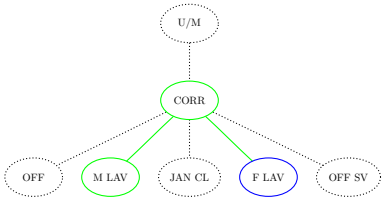
- p, the largest frequent subgraph present in the partial graph that has atleast one vertex in common with G (if found). p is also removed from the set P. If no such p could be found, it returns the empty graph $\emptyset$.

1: **for all** $p \in P$ **do**
2: **if** HasVertexInCommon$(G, p) \wedge p \not\subseteq \bigcup_{i=1}^{n} C_i$ **then**
3: $P \leftarrow P \setminus \{p\}$
4: **return** p
5: **end if**
6: **end for**
7: **return** $\emptyset$

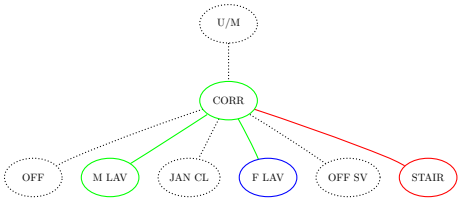


(a) Original state

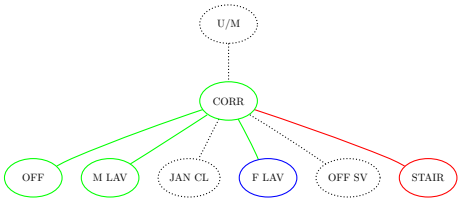
(b) First prediction



(c) Second prediction



(d) Third prediction



(e) Fourth prediction

Fig. 6: The evolution of a predicted graph with four consecutive predictions. The dashed lines are the unknown true graph. The blue nodes and edges are correspond to the initial input graph, green represents a prediction that exists in the true graph whereas red represents a predicted node or edge absent in the actual true graph.

Algorithm 3 Find most likely graph edit operation

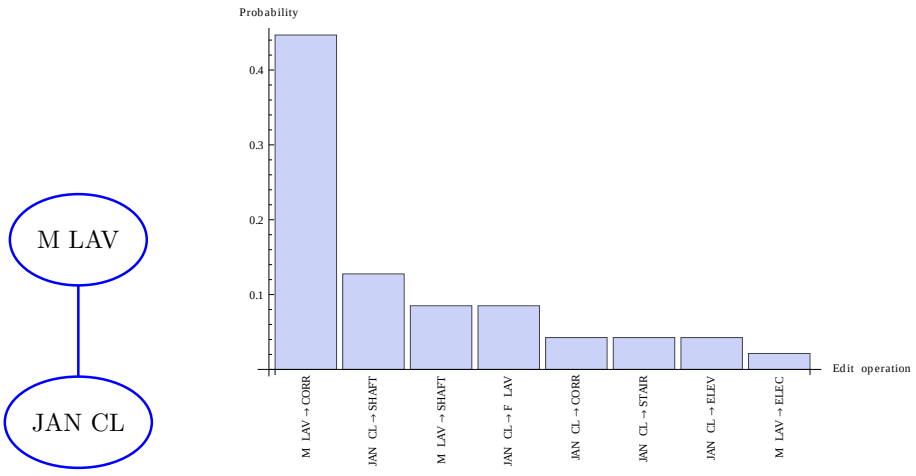
Input:

- G, a "small" graph, one subgraph from the output of the graph splitting.
- $\mathcal{D}$, the graph database

Output:

- G', the graph which is the result of performing the optimal edit operation upon G

for $x \in \mathcal{D}$ **do**
  if $G \subseteq x$ **then**
    for $G' \in B(G,1) \wedge G' \subseteq x$ **do**
      {Every G' corresponds to some valid edit operation upon G (that is, both G and G' are contained in this specific graph x).}
      $\phi(x,G) \leftarrow \phi(x,G) + 1$
    end for
  end if
end for
return $\arg\max_{x \in B(G,1)} \phi(x,G)$



(a) Partial graph

(b) Probability distribution

Fig. 7: The discrete probability distribution for the edit operations of a partial graph. Given the partial graph in (a), vertex addition hypotheses are shown on the x-axis in (b), with corresponding probabilities.

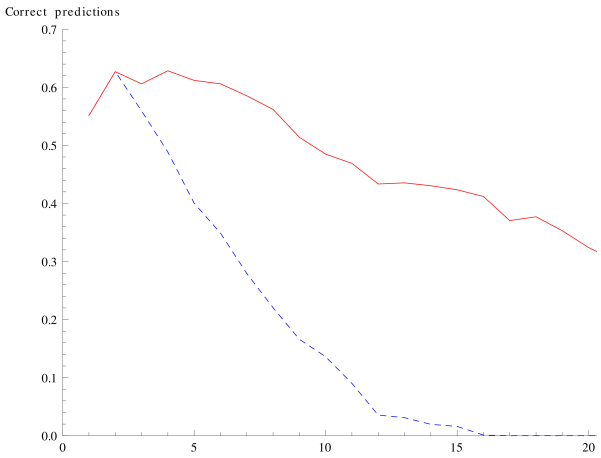


Fig. 8: Comparsion between the two prediction methods over 40,000 partial input graphs. The naive algorithm (*Method I*) is shown as the blue dashed line, and the prediction algorithm with graph splitting (*Method II*) as a solid red line.

As another example, the input graph in figure 7a results in the discrete probability distribution shown in figure 7b. Since this partial graph consists of only two vertices, the only edit operations considered are those of adding a new vertex. On the horizontal axis, the different edit operations are shown as $A \rightarrow B$, where A is some existing vertex of the partial graph and where B is the vertex which should be added and connected to A. Note that edit operations with a probability below 0.02 are not shown. In this case, A can only take the values of janitor closet "JAN CL" or male lavatory "M LAV". Note that as expected the *corridor* "CORR" vertex has the highest probability of being connected to another vertex by a large margin.

4.2 Quantitative evaluation

We have compared the results of the two prediction methods. To measure the performance of the algorithms for varying graph sizes, we have selected 2000 partial graphs randomly from the dataset, for each graph size between one and 20. In total 40000 different partial graphs were processed. The selection process works as follows. First we pick a random graph from the dataset $\mathcal{D}$. Then for a given graph size $m = \{1...20\}$, we pick at random m connected vertices which form an input graph. Then, we iterate this process until 2000 partial graphs are selected. Finally, the graphs from which random partial graphs were picked are excluded from the training dataset (multiple partial graphs may come from the same graph).

We counted the number of correct edit operations predicted by each algorithm over the test set, and divided by the total number of partial graph predictions to get a percentage of correct predictions (shown in figure 8). The naive

algorithm (*Method I* in section 3.2) is shown in dashed blue and the prediction algorithm with graph splitting (*Method II* in section 3.3) in red. For smaller graph sizes, their performance is almost equivalent. However for larger graphs, the performance of the naive algorithm decreases dramatically compared to the algorithm with splitting. The naive algorithm must compute support for edit operations on the *whole* graph, and is therefore subject to data sparsity and overfitting as the graph prediction size increases. Method II, however, leverages graph splitting and frequent subgraph extraction to focus on the *functional* components of the graph. This not only prunes the hypothesis space, but also enables greater predictive power through small functional groups, which have more substantial support in the dataset.

5 Conclusion and Future Work

In this paper we have provided an initial analysis of a large real-world indoor topological database. We have shown experimentally that the presented methods predict indoor topologies accurately. To the best of our knowledge, no previous work exists on analyzing and using a large real-world floorplan database for predicting indoor topologies. Furthermore, we have shown that indoor topologies consists of functional smaller parts which in turn can be used to develop methods with better prediction results. The reason for this is such methods capture the rationale behind man-made indoor spaces.
Following this work, we expect a large interest in developping the data-driven methods on indoor environments. We have yet to exploit the rich set of information offered by such datasets.

Future work consists of modeling the number of room types, extending the database with data from other environments such as KTH campus, making use of the metric coordinates in the data to have richer predictions and investigate how the predictions generalize for different locations.

6 Acknowledgements

The authors thank Emily Whiting, Seth Teller and the RVSN group at MIT CSAIL[1] for their helpful pointer to acquire the dataset. We are also grateful to John Folkesson and Kristoffer Sjöö for their feedback on an early draft of the paper. This work was supported by the EU FP7 project CogX.

[1] `http://rvsn.csail.mit.edu`

Bibliography

[1] 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. doi: 10.1016/j.imavis.2010.01.015. URL `http://www.pronobis.pro/publications/pronobis2010imavis`.

[2] A. Ranganathan. Pliss: Detecting and labeling places using online change-point detection. In *Proceedings of Robotics: Science and Systems*, Zaragoza, Spain, June 2010.

[3] Oscar Mozos, Cyrill Stachniss, Axel Rottmann, and Wolfram Burgard. Using AdaBoost for place labeling and topological map building. In Sebastian Thrun, Rodney Brooks, and Hugh Durrant-Whyte, editors, *Robotics Research*, volume 28, pages 453–472–472. Springer Berlin / Heidelberg, Berlin, Heidelberg, 2007. ISBN 978-3-540-48110-2.

[4] E. Whiting, J. Battat, and S. Teller. Topology of urban environments. In *Proc. of the Computer-aided architectural design futures (CAADFutures)*, pages 115–128, 2007.

[5] Gabriel Valiente. Efficient algorithms on trees and graphs with unique node labels. In *Applied Graph Theory in Computer Vision and Pattern Recognition*, pages 137–149. 2007.

[6] H Bunke and C Allermann. Inexact graph matching for structural pattern recognition. *Pattern Recognition Letters*, 1:245–253, 1983.

[7] Yiping Ke, James Cheng, and Jeffrey Xu Yu. Efficient discovery of frequent correlated subgraph pairs. In *Proceedings of the 2009 Ninth IEEE International Conference on Data Mining*, ICDM '09, pages 239–248, Washington, DC, USA, 2009. IEEE Computer Society.

[8] Xifeng Yan and Jiawei Han. gspan: Graph-based substructure pattern mining. In *Proceedings of the 2002 IEEE International Conference on Data Mining*, ICDM '02, pages 721–, Washington, DC, USA, 2002. IEEE Computer Society.