

Relational Object Maps for Mobile Robots

Benson Limketkai and Lin Liao and Dieter Fox

Department of Computer Science and Engineering

University of Washington

Seattle, WA 98195

Abstract

Mobile robot map building is the task of generating a model of an environment from sensor data. Most existing approaches to mobile robot mapping either build topological representations or generate accurate, metric maps of an environment. In this paper we introduce relational object maps, a novel approach to building metric maps that represent individual objects such as doors or walls. We show how to extend relational Markov networks in order to reason about a hierarchy of objects and the spatial relationships between them. Markov chain Monte Carlo is used for efficient inference and to learn the parameters of the model. We show that the spatial constraints modeled by our mapping technique yield drastic improvements for labeling line segments extracted from laser range-finders.

1 Introduction

Building maps of indoor spaces is an extremely important task in mobile robotics. Over the last years, the SLAM (simultaneous localization and mapping) community has made tremendous progress in the development of efficient and highly accurate map building techniques. Most of these techniques focus either on capturing the metric layout of an environment with high accuracy [Eliazar and Parr, 2003; Konolige *et al.*, 2005] or on representing the topological structure of an environment [Kuipers, 2000]. While metric maps have the advantage of being well suited for robot navigation tasks, they are typically unstructured and contain no information about the different types of places or objects in an environment. Topological approaches, on the other hand, are more compact and expressive in that they describe significant places in an environment. However, these techniques often ignore valuable metric information and they still lack the ability to describe individual objects in an environment.

Our goal is to build on the progress made in metric mapping and to enhance metric maps with semantic information about different objects in an environment. Object-based representations have much higher expressive power in that they combine metric accuracy with a semantic description of an environment. In addition to allowing more natural interfaces between humans and robots (“take the third door on the right”), they enable robots to perform better reasoning, especially about dynamic objects such as doors.

Recently, Anguelov and colleagues made important steps toward building such maps by developing techniques for detecting and labeling objects in metric maps. In [Anguelov *et al.*, 2002], they showed how to learn models of non-stationary, free-standing objects such as rectangular or round boxes. To overcome the limitations of this technique, they then developed a generative Bayesian approach for detecting doors and walls in a hallway environment [Anguelov *et al.*, 2004]. While this approach yields good results for labeling line segments extracted from laser range-scans, it has limited capabilities in modeling context information. Even though the approach can express information such as all doors share the same width and color, it can not model more complex, spatial relationships between objects. As [Murphy *et al.*, 2003] showed for scene analysis in computer vision, such context information can be extremely useful for distinguishing between different types of objects. In mobile robot map building, for example, an object standing in the middle of a hallway cannot be a door, even if it has the same width and color as other doors in the hallway.

In this paper, we show how to reason about the appearance of and spatial relationships between objects so as to build metric maps with object representations. We denote the resulting mapping paradigm *Relational Object Maps*, RO-Maps for short. RO-Maps build on Relational Markov Networks (RMN) to represent hierarchical structure and the relationships between objects. RMNs are undirected graphical models that are learned discriminatively [Taskar *et al.*, 2002]. They provide an extremely flexible framework for describing and reasoning about relations. Our RO-Maps detect objects such as doors and walls by a combination of features, including appearance features such as the width of a door, neighborhood features describing what type of objects are next to each other, and spatial features such as the indentation of a door relative to the wall. In our experiments we show that the parameters of RO-Maps can be learned from a set of hallways and then be successfully applied to labeling the objects in a hallway of a different environment.

This paper is organized as follows. After providing background on RMNs, we will show how RO-Maps model object appearance and the spatial relationships between them. Inference and parameter learning will be discussed in Section 3. Experimental results are presented in Section 4, followed by conclusions and a discussion of future work.

2 Relational Map Building

This section describes RMNs and several extensions that make them applicable to the problem of mobile robot map building. We denote the resulting framework RO-Maps, short for relational object maps.

2.1 Relational Markov Networks

We will now briefly review the basic ideas of relational Markov networks, more detailed information can be found in [Taskar *et al.*, 2002]. RMNs are an extension of Conditional Random Fields (CRFs), which are undirected graphical models that were developed for labeling sequence data [Lafferty *et al.*, 2001]. CRFs are discriminative models that have been shown to out-perform generative approaches such as HMMs and Markov random fields in areas such as natural language processing [Lafferty *et al.*, 2001] and computer vision [Kumar and Hebert, 2003]. RMNs extend CRFs by providing a relational language for describing clique structures and enforcing parameter sharing at the template level. Thereby RMNs are an extremely flexible and concise framework for defining features and relationships that can be used in our map building context.

Specifically, an RMN consists of three parts: a *schema* \mathcal{E} for the domain, a set of *relational clique templates* \mathcal{C} , and corresponding *potentials* Φ . The schema \mathcal{E} specifies the set of *classes* (*i.e.*, entity types) and the attributes in each class. In the map building context, the classes are types of objects, such as walls or doors, and the attributes describe their physical properties. An *instantiation* \mathcal{I} of a schema specifies the set of entities for each class and the values of all attributes for each entity. In our case, \mathcal{I} is an RO-Map consisting of line segments extracted from laser range-scans. A *relational clique template* $C \in \mathcal{C}$ is similar to a query in a relational database. It selects tuples from an instantiation \mathcal{I} ; the query result is denoted as $C(\mathcal{I})$. In an RO-Map, for example, a clique template can select all wall or door objects on one side of a hallway.

Each clique template C is associated with a potential function $\phi_C(\mathbf{v}_C)$ that maps values \mathbf{v}_C of variables in the clique to non-negative real numbers. Using a log-linear combination of feature functions to define these potentials, we get the following representation: $\phi_C(\mathbf{v}_C) = \exp\{\mathbf{w}_C^T \cdot \mathbf{f}_C(\mathbf{v}_C)\}$, where $\mathbf{f}_C()$ defines a feature vector for C and \mathbf{w}_C^T is the transpose of the corresponding weight vector. For instance, a feature could be the indentation of a door, defined on a clique template selecting a door and wall objects on one side of a hallway.

For a specific instantiation \mathcal{I} , an RMN defines a conditional distribution $p(\mathbf{y}|\mathbf{x})$ over labels \mathbf{y} given observations \mathbf{x} . To compute the conditional distribution of a label vector \mathbf{y} , the RMN generates an *unrolled* Markov network, in which the nodes correspond to the entities. The cliques of the unrolled network are built by applying each clique template $C \in \mathcal{C}$ to the instantiation, which can result in the generation of several cliques per template. All cliques that originate from the same template must share the same weights \mathbf{w}_C . The resulting cliques factorize the conditional distribution as

$$p(\mathbf{y} | \mathbf{x}) = \frac{1}{Z(\mathbf{x})} \prod_{C \in \mathcal{C}} \prod_{\mathbf{v}_C \in \mathcal{C}} \phi_C(\mathbf{v}_C) \quad (1)$$

$$\begin{aligned} &= \frac{1}{Z(\mathbf{x})} \prod_{C \in \mathcal{C}} \prod_{\mathbf{v}_C \in \mathcal{C}} \exp\{\mathbf{w}_C^T \cdot \mathbf{f}_C(\mathbf{v}_C)\} \quad (2) \\ &= \frac{1}{Z(\mathbf{x})} \exp\{\mathbf{w}^T \cdot \mathbf{f}\}, \quad (3) \end{aligned}$$

where the normalizing partition function $Z(\mathbf{x}) = \sum_{\mathbf{y}} \prod_{C \in \mathcal{C}} \prod_{\mathbf{v}_C \in \mathcal{C}} \phi_C(\mathbf{v}_C)$. (3) follows by moving the products into the exponent and combining all summations into \mathbf{w} and \mathbf{f} .

For efficient inference in RMNs, [Taskar *et al.*, 2002] suggest using (loopy) belief propagation. However, because of the extensions described below, we cannot apply belief propagation directly. Instead, we will show how to perform efficient inference and learning using MCMC.

2.2 Relational Object Maps

In the most general form, RO-Maps represent an environment by a hierarchical collection of objects and relations between them. RO-Maps estimate the types of objects under the assumption that their locations are known. This assumption is justified by the progress made by the SLAM community, which focuses on the problem of concurrently estimating the pose of a robot and the locations of sensor measurements. Such techniques generate well aligned laser range-scans even for large scale environments (*e.g.*, [Eliazar and Parr, 2003; Konolige *et al.*, 2005]).

We introduce an extension of the relational framework of RMNs to reason about physical objects in RO-Maps. The RO-Map of a specific environment is modeled by an unrolled Markov network. Each node in such a network corresponds to an object or geometric primitive, and the links between nodes represent relations between these objects.

Objects: The basic building blocks of RO-Maps are geometric primitives such as line segments extracted from sensor data. These primitives typically describe objects such as doors or wall segments. More complex objects can be generated from existing ones by a process we call *physical aggregation*. This process first selects all objects with a certain property (just like relational clique templates do in RMNs), and then computes the parameters of the aggregated object. For example, physical aggregation generates a wall object by selecting all wall segments on one side of a hallway, followed by the computation of the line parameters and color distribution describing the wall. The left and right wall of a hallway can be aggregated into a hallway object, which has parameters such as orientation and width.

Relations between objects can be spatial, such as “distance from each other”, or appearance based, such as “similar color”. Relations can be defined on basic objects and objects generated by (multiple) physical aggregations. Relational clique templates of RMNs can be used to define RO-Map relations. For example, to consider the similarity between the widths of the doors in a hallway, RO-Maps use a clique template C that first selects all objects that are labeled as doors (such selections are defined via SQL queries on the objects [Taskar *et al.*, 2002;

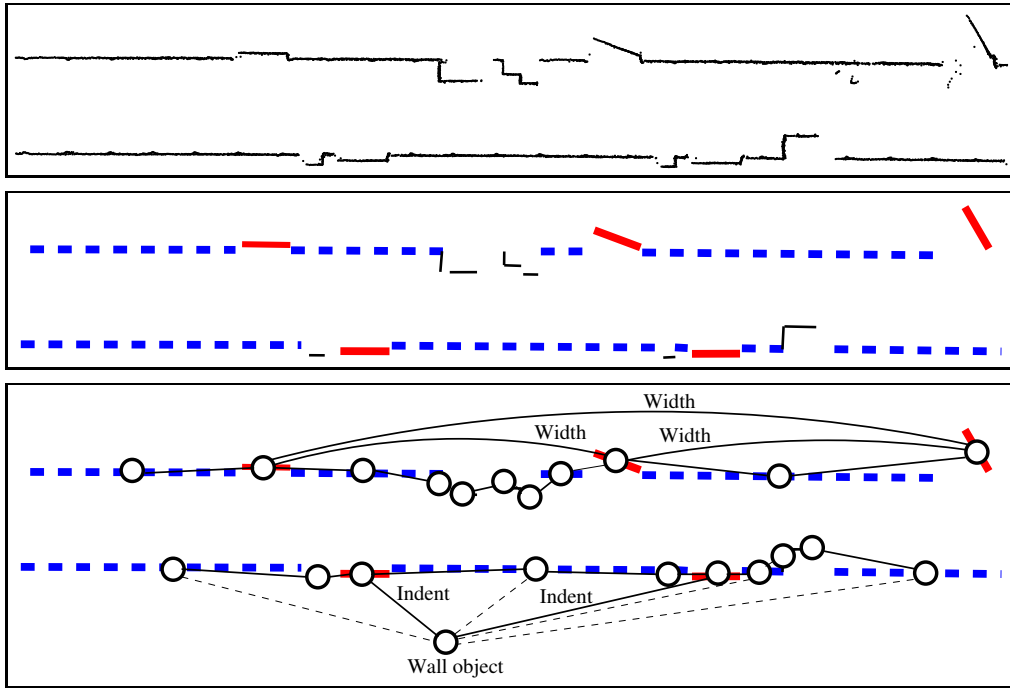


Figure 1: Upper graph: Endpoints of laser range-scans observed in a hallway. The points line up very accurately due to the position correction performed by a scan matching procedure. Middle graph: Line segments extracted from the scans. Wall segments are shown as dashed lines, doors as solid lines, others as thin black lines. Lower figure: Markov network with example cliques generated for this specific labeling. Thin solid lines define neighborhoods. The doors in the upper part of the hallway are linked by the door variance clique template. The wall object is generated by physical aggregation (dashed lines). It is linked to doors by the door indentation clique template (solid lines).

Liao *et al.*, 2005]). The widths of the selected doors define the value vector \mathbf{v}_C of this clique, which is then fed into the feature function $\mathbf{f}_C()$ that computes the variance. The potential of the clique, $\phi_C(\mathbf{v}_C)$, is given by $\exp\{w_C \cdot \mathbf{f}_C(\mathbf{v}_C)\}$, where w_C is the feature weight.

RO-Maps are illustrated in Figure 1. The upper plot shows laser range-scans observed by a robot traveling down a hallway; the middle plot shows the labeled line segments extracted from these scans; and the lower plot shows the nodes in the corresponding RO-Map along with some of the undirected links generated by the clique templates for this labeling. Among others, the figure shows two cliques generated by the template that measures the indentation of doors. The template generates pairs of doors and the corresponding wall object. The potential of the clique is then based on the distance of the door from the line representing the wall. More details on features for detecting doors and walls will be given in the experimental results section.

3 Inference and Learning

3.1 Inference: Labeling objects in RO-Maps

The goal of inference is to estimate the labels (types) of the objects represented in an RO-Map. Inference in RO-Maps is more complicated than in regular RMNs. This is due to the fact that the clique structure of the Markov networks generated by RO-Maps depends on the unknown labels of objects, and thus changes during inference. Since it is not clear how (loopy) belief propagation can be applied to such *label-*

specific cliques, we use Gibbs sampling, an MCMC technique, for inference [Gilks *et al.*, 1996]. Whenever the label of an object is changed during sampling, we determine all (aggregated) objects and cliques that could be affected by this change and re-compute their parameters and potentials. To initialize a Markov chain, we randomly assign labels to the basic objects in the RO-Map. Based on these labels, we generate all aggregate objects and their parameters. Finally, the relational clique templates generate the cliques of the unrolled, fully instantiated Markov network.

At each MCMC step, we update the label of a randomly selected object by sampling from the conditional distribution

$$p(y_k | \mathbf{y}_{-k}, \mathbf{x}, \mathbf{w}) \propto \exp\{\mathbf{w}^T \cdot \mathbf{f}(\mathbf{x}, \mathbf{y}_{-k} \cup y_k)\} \quad (4)$$

where k is the index of the object, y_k is one of its possible labels, and \mathbf{y}_{-k} are the labels for the other objects. To compute the probabilities of the different labels y_k , we update the cliques and the parameters of aggregated objects involving the object k . Even though this process can become inefficient for highly complex RO-Maps, inference on our current test hallways is very efficient, as we will describe in Section 4. We expect further efficiency gains by more elaborate sampling techniques.

3.2 Learning: Determining feature weights

The parameters of RO-Maps are the weights \mathbf{w} of the features that define the clique potentials. The key idea of our technique is to learn these parameters from a set of labeled environments, and then apply the learned weights when the robot explores a new environment.

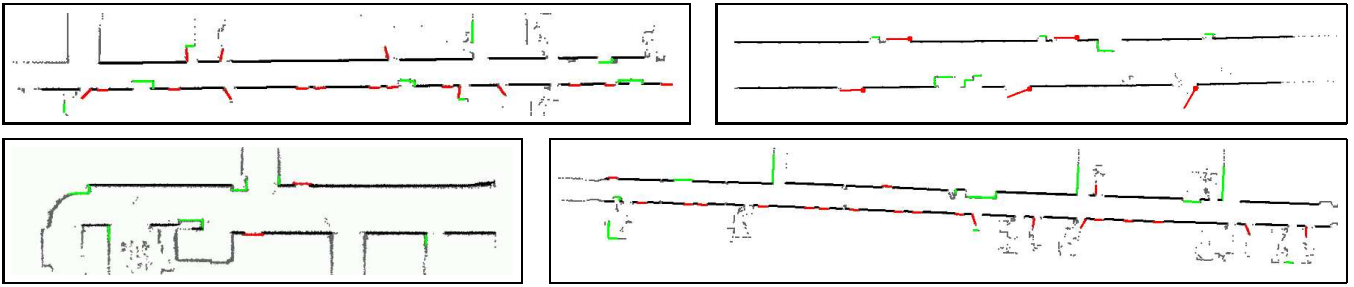


Figure 2: Examples of hallways used for learning and testing. Line segments were labeled manually as Door, Wall, or Other.

The details of our learning technique are beyond the scope of this paper, more information is given in [Liao *et al.*, 2005]. In a nutshell, the weights are learned by minimizing the *negative* log-likelihood of the labeled training data collected in N environments:

$$L(\mathbf{w}) \equiv \sum_{j=1}^N -\log p(\mathbf{y}_j | \mathbf{x}_j, \mathbf{w}) + \frac{\mathbf{w}^T \mathbf{w}}{2\sigma^2} \quad (5)$$

Here, j is the index of map m_j , and \mathbf{y}_j are the labeled objects in the j -th map. The rightmost term avoids overfitting by imposing a zero-mean, Gaussian shrinkage prior on each component of the weight vector [Taskar *et al.*, 2002]. Since (5) is convex, we can determine the global minimum using a quasi-Newton method, where the value and gradient of $L(\mathbf{w})$ are determined by MCMC sampling [Liao *et al.*, 2005].

The outcome of the learning procedure is a set of feature weights that best represent the appearance and spatial constraints of the training maps by maximizing the classification rate. RO-Maps inherit the important benefit from RMNs of automatically enforcing parameter sharing between (aggregated) objects and relations of the same type. This “smoothing” over instances of classes provides a very powerful tool for dealing with sparse training data.

4 Implementational Details and Experiments

We evaluated RO-Maps on the task of labeling line segments extracted from laser range-finder scans collected by a robot traversing a hallway. The data sets were taken from the Radish robotics data set repository [Howard and Roy, 2003]. We took two to three hallways from five different environments each and manually labeled the extracted and aligned line segments into the three categories Door, Wall, and Other. Line segments were extracted using the technique described in [Gutmann and Konolige, 1999]. Some of the test hallways are shown in Figure 2. To perform the labeling, we relied on our experience in typical layouts of hallways. In several cases, we were not sure about the labeling or were only able to label doors based on the fact that we knew the real buildings represented by the maps.

4.1 Features

To label the line segments in the hallways, we implemented various features. The features were represented by the log-likelihood of the measured values under a generative model. This approach allows our technique to automatically take variability between different features into account.

Physical aggregation Wall objects are the only physically aggregated objects in our current system. Wall objects are lines that are generated from line segments that are labeled Walls and are on the same side of a hallway (see Figure 1). The Wall object parameters can be computed efficiently from the parameters of the line segments.

Local features describe appearance properties of objects. In RO-Maps, such features are modeled by generating a clique for each object and the corresponding attribute. We use the length of line segments as a feature. The corresponding feature functions return the log-likelihood of a specific length given a Gaussian representing the distribution over lengths, conditioned on the object type. The means and variances of the Gaussians for Doors, Walls, and Others were estimated from the labeled maps. In our framework it is straightforward to add other features, such as the color of objects [Angelov *et al.*, 2004]. Unfortunately, the Radish data sets do not contain camera information.

Aggregated Wall objects have a feature that measures the alignment of the line segments that make up the wall object. Alignment is measured by the average distance of the line segments from the aggregated Wall line. This feature helps RO-Maps to label line segments as Walls only when they are well aligned with other Wall segments.

Neighborhood features describe which object types are located next to each other. For example, it is common to observe a Wall segment next to a Door, but uncommon to find many Doors in close proximity to one another. We compute a line segment’s neighborhood by determining all other segments with endpoints within 40 cm. Neighborhood features are then modeled by binary indicator functions on the possible label pairs (*e.g.*, “WW”, “WD”, “DO”).

Spatial features describe the relative position of objects. The indentation of a Door object is computed relative to the aggregated Wall object on the same side of the hallway. Indentation is given by the minimum distance between the Door line endpoints and the Wall line, thereby estimating indentation at the hinge point of a door. As with line width, the indentation feature function returns the log-likelihood under a Gaussian indentation model estimated from the training maps.

Unless an object is a Wall segment, it is highly unlikely

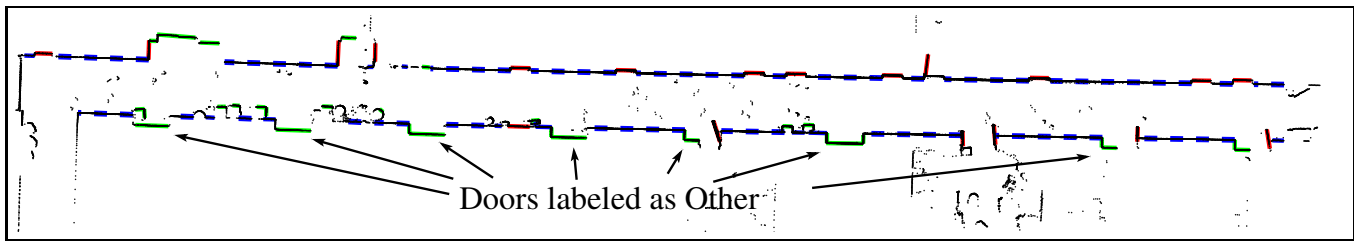


Figure 3: Hallway with the highest classification error. The doors in the lower half of the hallway are wide double doors with very large indentations. Since none of the other training environments contained such doors, our approach labeled them as Other. Some of the open double doors were detected as doors. All doors in the upper half and all walls (dashed lines) are labeled correctly.

to be both very close and well-aligned with a Wall object. RO-Maps detect Other objects by considering the distance and angle alignment of such objects relative to the closest Wall object. To compute the likelihood of specific distances and angles, we estimate discrete distributions from the training maps (these distributions are typically non-Gaussian).

Global features depend on multiple objects that are potentially far apart. In our experiments, we measure the similarity of the indentations of all doors in a hallway by computing their variance.

4.2 Experiments and Results

To test the performance of our learning method, we used 5-fold cross-validation on the five environments extracted from the Radish repository. To do so, the model parameters were trained on labeled hallways from four environments and then tested on the remaining one. Training on a set of 10 hallways took 2-3 hours. To generate the statistics needed for each iteration of the quasi-Newton optimization technique, we ran a Gibbs sampler for 1,000 iterations on each training map. The final 200 samples from each chain were used to compute differences between the sampled feature values and the feature values extracted from the labeled maps. These differences were then used to compute the relative, negative log-likelihood (5) and the gradient.

Testing, that is, labeling of the objects in an unknown map, was very efficient and took around one minute or less on average. For each environment, the labeling error rate was determined by comparing the most probable assignment to the manually generated label. Most probable assignments were extracted from the Markov chains by counting the different labels for each object. The label with the highest count was compared to the ground truth. To gauge the importance of different features for labeling the objects in these hallways, we varied the set of features we used.

| Environm. | Lengths | Lengths + Neighbors | All |
|-----------|---------|---------------------|-------|
| 1 | 62.6% | 88.5% | 90.7% |
| 2 | 58.7% | 63.0% | 93.5% |
| 3 | 59.0% | 79.2% | 89.7% |
| 4 | 51.8% | 96.5% | 97.7% |
| 5 | 60.0% | 68.5% | 77.9% |

Table 1: Average accuracy rates on maps using different features.

Table 1 summarizes the results. The numbers give the ac-

curacy rates of the labels, averaged over the two hallways of each test environment. The leftmost result column gives the accuracy rates achieved when using only the lengths of line segments to distinguish between the different object types. The next column summarizes results obtained when considering length and neighborhood features. The resulting models are similar to standard HMMs or MRFs in that they consider local appearance features (length) and connect neighboring objects. As can be seen, object neighborhood improves the results significantly. Furthermore, these results indicate that the labeling task is by far not trivial. The last column presents the accuracy rates achieved when using all features discussed in the previous section. The results show that the additional consideration of spatial constraints such as door indentation improves the classification performance by a large margin.

Figure 3 shows the hallway with the highest classification error (environment 5). As can be seen, most of the doors along the lower wall of the hallway are mislabeled as Other. However, this is not surprising given that the indentation of these doors is far larger than in the other training maps. Furthermore, most of these double doors are closed, which results in very long line segments. Thus, the labels are very reasonable, given the training data available to the approach.

| Truth | Inferred labels | | |
|-------|-----------------|------|-------|
| | Wall | Door | Other |
| Wall | 221 | 5 | 8 |
| Door | 1 | 122 | 21 |
| Other | 10 | 12 | 93 |

Table 2: Confusion matrix (counts).

Table 2 shows the confusion matrix summarized over all test environments using all features. The table shows that most errors are due to confusion between Others and Walls, and Others and Doors. Walls and Doors are rarely confused. The overall accuracy rate is 88.4%. This is a very encouraging result, especially since several maps contained objects for which even humans were uncertain about the correct labeling.

It should be noted that [Anguelov *et al.*, 2004] achieved relatively high accuracy using a hierarchical Bayesian approach without spatial constraints. This is due to the fact that they trained model parameters in the same environment and took visual color information into account. Adding such appearance information to RO-Maps is straightforward and we expect strong improvements from such additional information.

5 Conclusions and Future Work

We introduced relational object maps for mobile robot mapping. RO-Maps use relational, discriminative techniques to generate metric maps with semantic object representations. RO-Maps build upon and extend Relational Markov Networks so as to reason about the appearance of and the spatial constraints between objects in an environment. In order to distinguish between different types of objects, RO-Maps can incorporate a variety of features, ranging from global, such as the width of all doors in a hallway, to spatial, such as the indentation of a door. To handle the changing structure of the Markov networks instantiated by RO-Maps, we showed how to perform inference and parameter learning using MCMC. During learning, RO-Maps benefit from automatic parameter sharing provided by relational clique templates. This allows the models to be trained from relatively sparse training sets.

Experiments in five different environments demonstrate that RO-Maps are able to achieve high accuracy in labeling objects in previously unknown hallways. We show that the accuracy is improved drastically by taking spatial constraints between objects into account. These results are particularly encouraging given that objects are labeled based on laser range-data only.

RO-Maps provide an extremely flexible framework for combining metric accuracy with the representation of individual objects and structures in an environment. We believe that RO-Maps are an important step toward more expressive, semantic representations of environments. However, the work presented here is only the first step in this direction, laying the foundation for future research projects.

Our current implementation relies on line segments extracted from laser range-scans as the building blocks of RO-Maps. As shown by [Anguelov *et al.*, 2004], color information extracted from cameras strongly improves the performance of object classification. The flexibility of our framework allows us to integrate such additional appearance information without modifications to the underlying RO-Map concept. We observed significant improvements in initial experiments with a camera pointed toward the walls. We additionally plan to improve line segmentation by incorporating segmentation into the RO-Map inference process. Thereby, high-level information provided by the RO-Map can guide range-scan segmentation.

The RO-Maps presented here model objects in individual hallways only. A key next step will be the extension to more object classes and multiple, connected hallways, rooms, and open spaces. The RO-Map framework enables the definition of high-level structures such as hallways or rooms using the notion of physical aggregation. We are currently developing hierarchical representations and inference techniques that label both high-level structures and individual objects within these structures.

6 Acknowledgments

The authors would like to thank Jeff Bilmes for fruitful discussions during the early stages of this project. This research is sponsored in part by the National Science Foundation (CA-REER grant number 0093406).

References

- [Anguelov *et al.*, 2002] D. Anguelov, R. Biswas, D. Koller, B. Limketkai, S. Sanner, and S. Thrun. Learning hierarchical object maps of non-stationary environments with mobile robots. In *Proc. of the Conference on Uncertainty in Artificial Intelligence (UAI)*, 2002.
- [Anguelov *et al.*, 2004] D. Anguelov, D. Koller, E. Parker, and S. Thrun. Detecting and modeling doors with mobile robots. In *Proc. of the IEEE International Conference on Robotics & Automation (ICRA)*, 2004.
- [Eliazar and Parr, 2003] A. Eliazar and R. Parr. DP-SLAM: Fast, robust simultaneous localization and mapping without predetermined landmarks. In *Proc. of the International Joint Conference on Artificial Intelligence (IJCAI)*, 2003.
- [Gilks *et al.*, 1996] W.R. Gilks, S. Richardson, and D.J. Spiegelhalter, editors. *Markov Chain Monte Carlo in Practice*. Chapman and Hall/CRC, 1996.
- [Gutmann and Konolige, 1999] J.S. Gutmann and K. Konolige. Incremental mapping of large cyclic environments. In *Proc. of the IEEE International Symposium on Computational Intelligence in Robotics and Automation (CIRA)*, 1999.
- [Howard and Roy, 2003] A. Howard and N. Roy. The robotics data set repository (radish), 2003. radish.sourceforge.net.
- [Konolige *et al.*, 2005] K. Konolige, D. Fox, C. Ortiz, A. Agno, M. Eriksen, B. Limketkai, J. Ko, B. Morisset, D. Schulz, B. Stewart, and R. Vincent. Centibots: Very large scale distributed robotic teams. In M. Ang and O. Khatib, editors, *Experimental Robotics: The 9th International Symposium*, Springer Tracts in Advanced Robotics (STAR). Springer Verlag, 2005.
- [Kuipers, 2000] B. Kuipers. The spatial semantic hierarchy. *Artificial Intelligence*, 119:191–233, 2000.
- [Kumar and Hebert, 2003] S. Kumar and M. Hebert. Discriminative random fields: A discriminative framework for contextual interaction in classification. In *Proc. of the International Conference on Computer Vision (ICCV)*, 2003.
- [Lafferty *et al.*, 2001] J. Lafferty, A. McCallum, and F. Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proc. of the International Conference on Machine Learning (ICML)*, 2001.
- [Liao *et al.*, 2005] Lin Liao, Dieter Fox, and Henry Kautz. Location-based activity recognition using relational Markov networks. In *Proc. of the International Joint Conference on Artificial Intelligence (IJCAI)*, 2005.
- [Murphy *et al.*, 2003] K. Murphy, A. Torralba, and W. Freeman. Using the forest to see the trees: A graphical model relating features, objects and scenes. In *Advances in Neural Information Processing Systems (NIPS)*, 2003.
- [Taskar *et al.*, 2002] Ben Taskar, Pieter Abbeel, and Daphne Koller. Discriminative probabilistic models for relational data. In *Proc. of the Conference on Uncertainty in Artificial Intelligence (UAI)*, 2002.