# 2D Semantic Mapping on Occupancy Grids

Ziyuan Liu[1,2,3], Dong Chen[1,2] and Georg von Wichert[2,3]

[1]Institute of Automatic Control Engineering, Technische Universität München, Munich, Germany
[2]Siemens AG, Corporate Research & Technologies, Munich, Germany
[3]Institute for Advanced Study, Techniche Universität München, Lichtenbergstrasse 2a, D-85748 Garching, Germany

## Abstract

In recent years, techniques for building metric maps of indoor environments have been intensely studied, and they perform very well in noumerous applications. Simultaneous Localization and Mapping (SLAM) methods produce globally consistent, metric maps of the explored environment. Although such maps describe how the environment looks like and can be used for navigation, there exist no abstracted semantic concepts that explain the environment on a higher level or in a more natural way (as we humans do), such as, what kind of structure and connectivity the environment possesses. In this paper, we propose a new probabilistic method to analyze the underlying semantic world model based on an occupancy grid map, which is generated by a standard SLAM process. Our approach simulates a Markov Chain that produces samples from the distribution of probable semantic world models given an input map. Experiments show that our approach is effective and correctly captures the uncertainty.

## 1 Introduction and related work

In recent years, techniques for building metric maps of indoor environments have been intensely studied, and they perform very well in many applications. Simultaneous Localization and Mapping (SLAM) methods produce a globally consistent, metric map of the explored environment. Although such maps describe how the environment looks like, there exist no abstracted semantic concepts that explain the environment on a higher level or in a more natural way (as we humans do), such as, what kind of structure and connectivity the environment possesses. Despite the accuracy of such metric maps, they are still just a huge unsorted matrix containing certain values like occupancy, intensity, etc.. Typical semantic concepts, such as rooms and corridors, or spatial relations like adjacency, connectivity via doors, or properties like rectangularity that – if known to be relevant to the given environment – could help to build the maps in the first place.

Although semantic robot mapping has not been as intensively studied as metric or topological mapping, some notable conbtributions have already been made. As an early predecessor of many semantic mapping approaches one might consider [11], which combined the grid-based and topological mapping to gain both of accuracy/consistency (metric) and efficiency (topological), the latter effectively by means of abstraction. Wolf and Sukhatme [14] proposed to use hidden Markov models and support vector machines to tackle the problem of terrain mapping and activity-based mapping. [10], [2] and [3] used semantic labels to annotate the places and regions explored by a mobile robot. Douillard et. al. proposed to use conditional random fields to build object-type maps of outdoor environments. Other examples of assigning semantic labels to perceived objects in the explored environments are presented in [7] and [13]. In addition, methods, which semantically annotate the structure of the environments, like traversable terrain in outdoor environments or walls, cellings and door in indoor environments, have also been proposed. Some remarkable examples are [8], [12], [6] and [9].

## 2 Problem description

Different from those methods mentioned above, we aim to construct a probabilistic generative model of the world around the robot, that is essentially based on abstract semantic concepts but at the same time allows to predict the continuous percepts that the robot obtains via its noisy sensors. This abstract model has a form similar to a scene graph, a structure which is widely used in computer graphics. The graph (see **Figure** 1d) in our case consists of rooms and doorways connecting the rooms and can be visualized as a classical floor plan (see **Figure** 1b). As a by-product, the connectivity information is represented as a topological map (see **Figure** 1c) which is of great help for navigation purpose.

The scene graph, i.e. the semantically annotated world state, is represented as a vector of parameters $W$ observed through the occupancy map $M$. In Bayesian framework we can use a maximum posterior approach to infer the most probable state $W^* \in \Omega$ from the space of possible worlds $\Omega$ given the map $M$.

$$W^* = \underset{W \in \Omega}{\operatorname{argmax}}\, p(W|M), \tag{1}$$

where

$$p(W|M) \propto p(M|W)p(W). \tag{2}$$

Here $p(W|M)$ is the posterior distribution of $W$ given a map $M$, and $p(W)$ is the prior specifying, which worlds $W$ are possible at all. $p(M|W)$ is the likelihood function

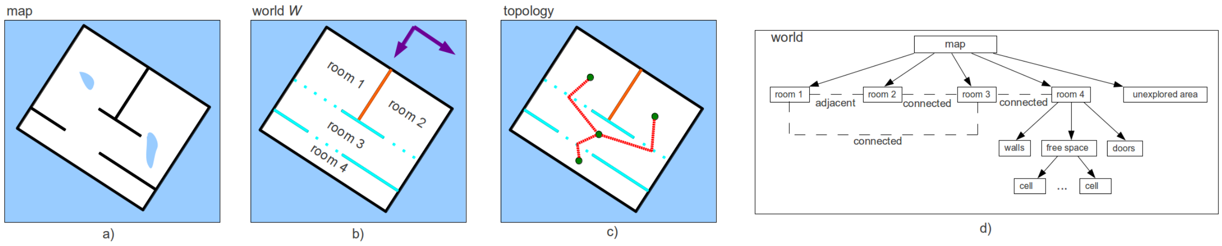


**Figure 1:** a) A simplified occupancy grid map: Unexplored area is drawn in blue, free space is drawn in white. Occupied area is drawn in black. b) A possible floor plan represented as a scene graph ($W$): The world is divided into four rooms and the corresponding unexplored area. Connectivity is given by the color of walls: the color cyan indicates *connected*, which means there is a door (cyan dotted) between two rooms; the color orange means *adjacent*, which means that two rooms are neighbor and do not connect themselves through a door; the color black stands for a boundary wall. The detected main orientations of walls are illustrated by violet arrows. c) The connectivity information represented as a topological map of world $W$: Green circles indicate room centers, and red lines connect the room centers and their corresponding doors. d) The semantic description of the world in form of the scene graph: Directed links connect nodes. The dashed lines indicate connectivity which is represented as a topological map in c). Like room 4, each room has three child nodes: walls, free space, and doors. Note that the lowest level of node in the tree structure is the grid cell that either belongs to a wall, free space or a door.

describing how probable the observed map $M$ is, given the different possible worlds represented by a parameter vector $W$. The actual semantic model is represented in the structure of the parameter vector $W$, while semantically relevant constraints go into the prior $p(W)$.

# 3 A generative model for occupancy grids

The prior $p(W)$ in (2) expresses a set of assumptions concerning the structured world based on context knowledge as follows:

1) a room has four walls and possesses a rectangular shape.
2) a room has at least one door, and a door is placed on a wall.
3) each cell in the map should only belong to one room.
4) walls of an indoor environment have two main orientations (see **Figure** 1b).

The prior $p(W)$ penalizes worlds that are not fully compliant with the above assumptions:

$$p(W) = \alpha_1 \times \alpha_2 \times \alpha_3 \times \alpha_4, \tag{3}$$

where $\alpha_1$, $\alpha_2$, $\alpha_3$ and $\alpha_4$ are the corresponding penalization terms for the point 1), 2), 3) and 4) of the prior information respectively.
Furthermore, for our generative model, we need to specify the likelihood function $p(M|W)$. Since $M$ is represented by an occupancy grid with statistically independent grid cells $c \in M$, we only need to come up with a model $p(c|W)$ for all cells at their locations $(x, y)$ in the map M:

$$p(M|W) = \prod_{c(x,y)\in M} p(c(x, y)|W). \tag{4}$$

For our model $p(c(x, y)|W)$, we first discretize the cell state $M(x, y)$ by classifying the intensity values into three classes $C_M(x, y)$ according to:

$$C_M(x, y) = \begin{cases} 2, & 0 \le M(x, y) \le h_o, \\ 1, & h_o < M(x, y) \le h_u, \\ 0, & h_u < M(x, y), \end{cases} \tag{5}$$

where $h_o$ and $h_u$ are the intensity thresholds for occupied, unexplored and free cells. Based on our world model $W$ we can also predict expected cell states $C_W(x, y)$ accordingly:

$$C_W(x, y) = \begin{cases} 2, & (x, y) \in S_w, \\ 1, & (x, y) \in S_u, \\ 0, & (x, y) \in S_f, \end{cases} \tag{6}$$

where $S_w, S_u$ and $S_f$ are the set of all the wall cells, unknown cells and free space cells in the world $W$ respectively. $p(c(x, y)|W)$ can then be represented in the form of a lookup-table.
In principle the likelihood $p(c(x, y)|W)$ plays the role of a sensor model. In our case it captures the quality of the original mapping algorithm producing the grid map (including the sensor models for the sensors used during the SLAM process), and could be learned from labeled training data.

# 4 Searching the solution space

For solving (1) we need to efficiently search the large and complexly structured solution space $\Omega$. Here we adopt the approach of [15], who propose a data driven Markov chain Monte Carlo (MCMC) technique for this purpose. The basic idea is to construct a Markov Chain that generates samples $W_i$ from the solution space $\Omega$ according to the distribution $p(W|M)$ after some initial burn-in time. One popular approach to construct such a Markov chain is the Metropolis-Hastings (MH) algorithm [1]. The Markov chain is constructed by sequentially executing state transitions (in our case from a given world state $W$ to another state $W'$) according to a transition distribution $\Phi(W'|W)$ of the sub-kernels. In order for the chain to converge to a given distribution, it has to be reversible and ergodic [1]. The MH algorithm achieves this by generating new samples in three steps. First a transition is proposed according to $\Phi(W'|W)$, subsequently a new sample $W'$ is generated by a proposal distribution $Q(W'|W)$, and then it is accepted with the probability $\lambda$.

$$\lambda(W, W') = \min\left(1, \frac{p(W'|M)Q(W|W')}{p(W|M)Q(W'|W)}\right) \quad (7)$$

The resulting Markov chain can be shown to converge to $p(W|M)$. However, the selection of the proposal distribution is crucial for the convergence rate. Here, we follow the approach of [15] to propose state transitions for the Markov chain using discriminative methods for the bottom-up detection of relevant environmental features (e.g. walls, doorways) and construct the proposals based on these detection results.

In order to realize the Markov chain in form of the Metropolis-Hastings algorithm, we arrange the kernels that alter the structure of the world as reversible pairs: ADD or REMOVE one room; SPLIT one room or MERGE two rooms; SHRINK or DILATE one room; ALLOCATE or DELETE one door.

**Figure** 2 shows an example of the four reversible MCMC kernel pairs. The world $W$ can transit to $W^{'}$, $W^{''}$, $W^{'''}$ and $W^{''''}$ by applying the sub-kernel REMOVE, MERGE, SHRINK and DELETE, respectively. By contrast, the world $W^{'}$, $W^{''}$, $W^{'''}$ and $W^{''''}$ can also transit back to $W$ using corresponding reverse sub-kernels.

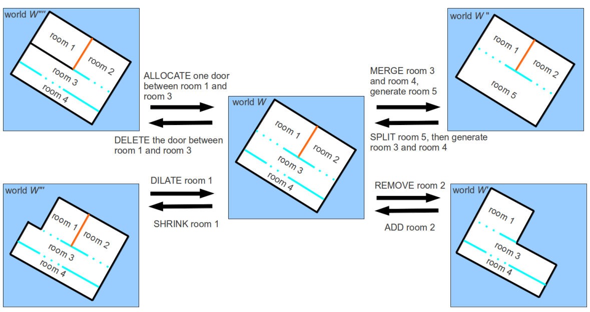


**Figure 2:** Reversible MCMC kernel pairs: ADD/REMOVE, SPLIT/MERGE, SHRINK/DILATE and ALLOCATE/DELETE.

The proposal probability $Q(W^{'}|W)$ and $Q(W|W^{'})$ describe how probable the world $W$ can transit to the world $W^{'}$ and $W^{'}$ can transit to $W$ respectively. They can be calculated in the following general form:

$$Q = p_k \times p_e, \quad (8)$$

where $p_k$ is the probability of choosing a certain sub-kernel to perform the transition. $p_e$ is the probability of choosing a certain element of the semantic world to change.

# 5 Experiments and discussions

Our approach was tested in several online and offline experiments. In online runs, we only utilize the robot position to propose new room candidates, since it is straightforward that there must exist a small free space around the current robot position. Thus, for the online runs, the discriminative method for the sub-kernel ADD is just generating a small rectangular region around the robot position. Then, these generated regions are locally adapted by other sub-kernels so as to better match the map. For offline runs, the information on robot position is not available, thus the discriminative method for proposing room candidates in ADD mainly focuses on finding room-like regions in the input map, and these regions are used as room candidates. Other sub-kernels stay the same as in online runs.

Once the semantic world model is built, the corresponding topological map can be easily computed based on the connectivity information. There are many ways to generate it, and in this paper, we just adopt a simple strategy which connects the room centers with the mid-point of their corresponding doors.

One example of an online run is depicted in 12 steps in **Figure** 3. In step 2), a room is proposed and added into the world. This room becomes bigger and bigger through many successful DILATE, as shown in step 3). Sub-kernels except ADD and DILATE tend to be rejected during this period, because they do not help to better explain the map. The same process happens again in step 4) to 6) and step 7) to 10). In step 11), a new room is generated through a successful SPLIT , and again, it is changed by DILATE to generate the final result shown in step) 12.

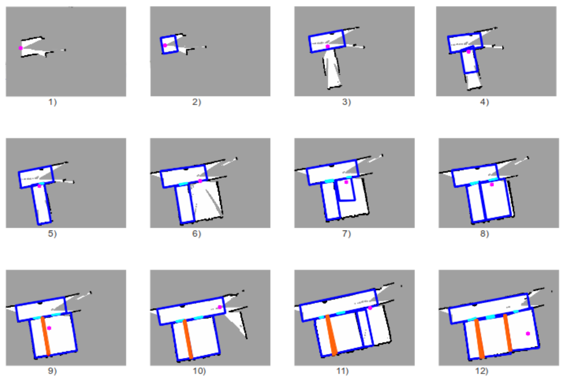


**Figure 3:** One example of an online run shown in 12 steps. Boundary walls (blue), adjacent walls (orange) and doors (cyan) of the world drawn into the classified map. The violet point depicts the robot position.

**Figure** 4 shows the final result of the online run explained above in some more detail. Here, part a) shows the final input occupancy map $M$. Part b) shows the classified map $C_M$ that is defined in (5), with the color black, gray and white indicating occupied, unexplored and free cells, respectively. Part c) visualizes the world state $W$ representing our structured semantic model. Here the colors black, gray, white and cyan show the wall, unknown, free and door cells respectively. The corresponding topological map is also depicted. In part d), boundary walls (blue), adjacent walls (orange) and doors (cyan) of the world $W$ are directly plotted onto the classified map, so as to give a more intuitive comparison. Another example of an online run is illustrated in **Figure** 6. **Figure** 7 demonstrates a result obtained offline for a much bigger environment.

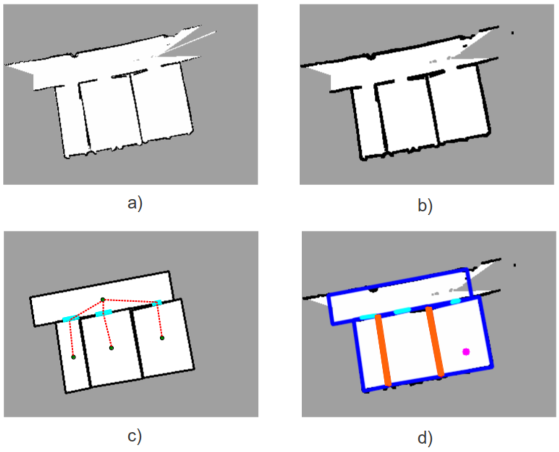


**Figure 4:** Final Result of the online run shown in **Figure** 3. a) An input map $M$ containing no furniture. b) The classified map $C_M$ with three intensity values (black=wall, grey=unexplored, white=free). c) The analyzed world $W$ (black=wall, gray=unknown, white=free, cyan=door) and the corresponding topological map (green circles=room centers). d) A direct comparison between the analyzed world and the input map: boundary walls (blue), adjacent walls (orange) and doors (cyan) of the world $W$ drawn into the classified map.

In order to quantitatively evaluate our approach, we compute $K(W, M)$, the *cell prediction rate* capturing the predictive power of the semantic world model $W$ and with respect to an input map $M$:

$$K(W, M) = \frac{\sum\limits_{c(x,y)\in M} l(c(x, y))}{t_M},$$

with

$$l(c(x, y)) = \begin{cases} 1, & C_M(x, y) = C_W(x, y), \\ 0, & \text{otherwise}, \end{cases} \tag{9}$$

where $t_M$ is the number of all grid cells in the map $M$. $c(x, y)$ indicates one grid cell located at the position $(x, y)$. $C_M(x, y)$ and $C_W(x, y)$ are previously defined in equation (5) and (6). $K(W, M)$ of the maps shown in this paper is summarized in **Table** 1.

Our current probabilistic world model does not consider furniture, therefore, compared with the map in **Figure** 4 which does not contain any furniture, the map in **Figure** 6 is more difficult to analyze due to the existence of furniture. This effect is clearly reflected in the cell prediction rate (98.7% vs. 93.8%). We will address this problem in the future by incorporating furniture and other objects as semantic concepts into the world model.

**Table 1:** Cell prediction rate $K(W, M)$.

| | **Figure** 4 d) | **Figure** 6 d) | **Figure** 7 d) |
|---|---|---|---|
| Percentage | 98.7% | 93.8% | 93.6% |

Our approach simulates a MCMC sampling which effectively draws samples from the underlying distribution of the semantic world model. In **Figure** 5, we illustrate this distribution by plotting 10000 samples together, which are obtained after the initial burn-in time. For illustration purposes, the walls of the semantic world model are drawn as very thin lines (see one sample in **Figure** 5a), so that the variance of the predicted wall postions can be seen as wall thickness in the overlay image (see **Figure** 5b). In contrast to a single sample, whose walls are partially too thin to be seen, the overlaid walls of 10000 samples almost fill all of the occupied cells of the input map, i.e. the thickness of the “real walls”. It is obvious that our approach correctly captures the uncertainty of the semantic world model. In comparison with the map of the online run shown in **Figure** 4a, the offline map in **Figure** 7a is much more complicated (mainly due to false positives caused by furniture or other objects), thus, the uncertainty of the semantic world model for this offline map is much bigger, as depicted in **Figure** 8b.

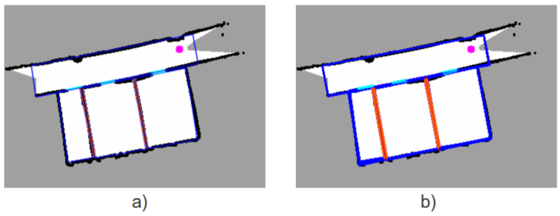


**Figure 5:** a) A single sample of the semantic world model. b) The distribution of the semantic world model for the map shown in **Figure** 4a built by 10000 samples. Note that the walls of each sample are drawn as very thin lines. The overlaid walls of 10000 samples almost fill the thickness of “real walls”, which makes it evident that our approach correctly captures the uncertainty of the semantic world model.

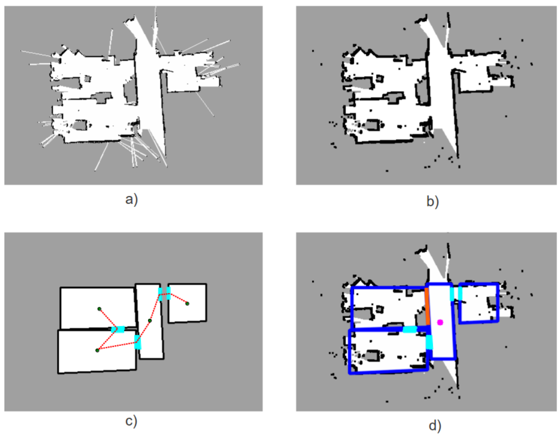


**Figure 6:** Another example of an online run using an input map which contains furniture. Color coding is the same as **Figure** 4.

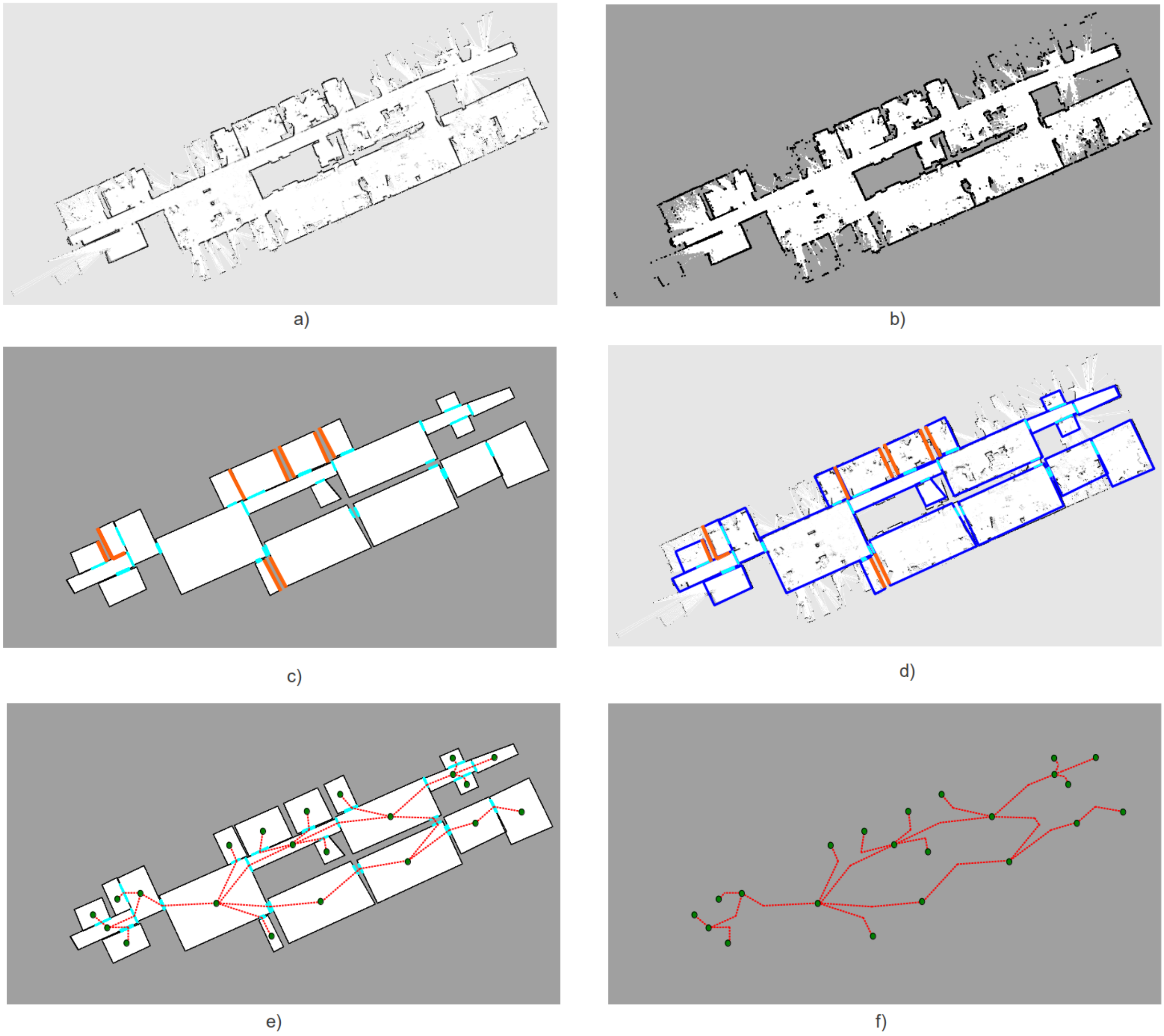


**Figure 7:** One example of an offline result. a) An occupancy grid map $M$ [5]. b) The classified map $C_M$ with three intensity values (black=wall, grey=unexplored, white=free). c) The analyzed world $W$ (black=boundary wall, orange=adjacent wall, gray=unknown, white=free, cyan=door). d) Boundary walls (blue), adjacent walls (orange) and doors (cyan) of the world $W$ drawn into the map. e) Generate the topological map using the connectivity information. f) The resulting topological map (green circles=room centers).

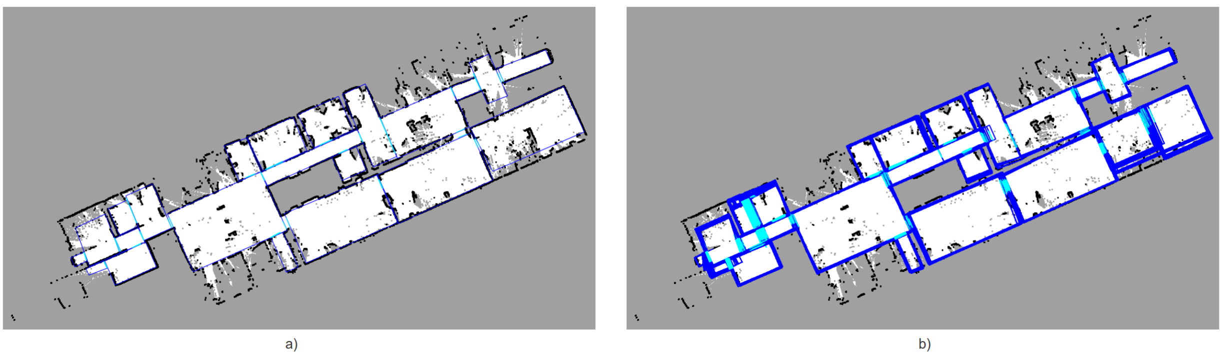


**Figure 8:** a) A single sample of the semantic world model. b) The distribution of the semantic world model for the map shown in **Figure** 7 a) built by 10000 samples. For illustration purpose, adjacent walls are not shown. Note that the walls of each sample are drawn as very thin lines.

In this paper, we make use of some basic assumptions concerning the environment to infer its semantic structure. The assumptions describe a common environment type being mainly recangular. However, our approach is in general not restricted to the specific assumptions used here, and it can be extended for other forms by incorporating new room types and corresponding discriminative methods.

Currently, our implementation is single-threaded. On an Intel Core i7 CPU with 2.0 GHz, the online mode can be finished within the runtime of the underlying SLAM process [4] which provides the input map for our algorithm. Of course, the computing speed largely depends on the size of the map that is to be analyzed (**Figure** 7, $1237 \times 672$, $\sim 1$ min).

# 6 Conclusions and future work

In this paper, we propose a new probabilistic method to extract the underlying semantic model of a given indoor environment based on an occupancy grid map that is generated by a standard SLAM process. Our approach implements a data driven Markov Chain Monte Carlo sampling procedure to draw samples from the distribution of possible semantic world models given this input map. Based on a few reasonable assumptions concerning the nature of indoor environments, we represent the domain in an abstracted, semantic and top-down manner, i.e. containing several rooms connected by doorways. Meanwhile, by using several discriminative environment feature detectors to propose Markov chain transisions, the method also incorporates a bottom-up path.

The method allows us to effectively sample from the probability distribution of possible interpretations of the continuous and noisy sensor observations. This is used here as a basis for finding the best explanation of the grid map in the maximum likelihood sense, and provides the robot with a structural understanding of its environment that can be used for a wide range of higher-level reasoning or communication purposes. At the current stage, our approach is limited to 2D cases, in which we build a classical floor plan from occupancy grids including semantic concepts. In the future, we will focus on extending our approach to 3D scenarios and enriching our world model with other semantic concepts.

# Acknowledgments

This work was accomplished with the support of the Technische Universität München - Institute for Advanced Study, funded by the German Excellence Initiative.

The work of one of the authors (Georg von Wichert) was partially made possible by funding from the ARTEMIS Joint Undertaking as part of the project R3-COP and from the German Federal Ministry of Education and Research (BMBF) under grant no. 01IS10004E

# References

[1] C. M. Bishop. *Pattern recognition and machine learning*. Springer, 2007.

[2] S. Friedman, H. Pasula, and D. Fox. Voronoi random fields: Extracting the topological structure of indoor environments via place labeling. In *Proc. of the International Joint Conference on Artificial Intelligence (IJCAI)*, 2007.

[3] N. Goerke and S. Braun. Building semantic annotated maps by mobile robots. In *Proceedings of the Conference Towards Autonomous Robotic Systems, Londonderry, UK*, 2009.

[4] G. Grisetti, C. Stachniss, and W. Burgard. Improved techniques for grid mapping with rao-blackwellized particle filters. *IEEE Transactions on Robotics*, 23(1):34–46, 2007.

[5] A. Howard and N. Roy. The robotics data set repository (radish), 2003.

[6] A.K. Krishnan and K.M. Krishna. A visual exploration algorithm using semantic cues that constructs image based hybrid maps. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1316 –1321, Oct. 2010.

[7] B. Limketkai, L. Liao, and D. Fox. Relational object maps for mobile robots. In *International Joint Conference on Artificial Intelligence*, volume 19, page 1471, 2005.

[8] A. Nüchter and J. Hertzberg. Towards semantic maps for mobile robots. *Robotics and Autonomous Systems*, 56(11):915–926, 2008.

[9] M. Persson, T. Duckett, C. Valgren, and A. Lilienthal. Probabilistic semantic mapping with a virtual sensor for building/nature detection. In *International Symposium on Computational Intelligence in Robotics and Automation*, pages 236 –242, June 2007.

[10] A. Pronobis, P. Jensfelt, K. Sjöö, H. Zender, G. M. Kruijff, O. M. Mozos, and W. Burgard. Semantic modelling of space. In *Cognitive Systems*, volume 8 of *Cognitive Systems Monographs*, pages 165–221. Springer Berlin Heidelberg, 2010.

[11] S. Thrun and A. Bücken. Integrating grid-based and topological maps for mobile robot navigation. In *Proceedings of the AAAI Thirteenth National Conference on Artificial Intelligence*, Portland, Oregon, 1996.

[12] Jingchen Tong, Dong Chen, Yan Zhuang, and Wei Wang. Mobile robot indoor semantic mapping using 3d laser scanning and monocular vision. In *8th World Congress on Intelligent Control and Automation (WCICA)*, pages 1212 – 1217, July 2010.

[13] Tingqi Wang and Qijun Chen. Object semantic map representation for indoor mobile robots. In *International Conference on System Science and Engineering (ICSSE)*, pages 309 –313, June 2011.

[14] D.F. Wolf and G.S. Sukhatme. Semantic mapping using mobile robots. *IEEE Transactions on Robotics*, 24(2):245–258, 2008.

[15] S.C. Zhu, R. Zhang, and Z. Tu. Integrating top-down/bottom-up for object recognition by data driven markov chain monte carlo. In *IEEE Computer Vision and Pattern Recognition (CVPR)*, 2000.