T-SLAM: Registering Topological and Geometric Maps for Robot Localization in Large Environments

F. Ferreira, I. Amorim, R. Rocha and J. Dias
Institute of Systems and Robotics
Department of Electrical and Computer Engineering
University of Coimbra, Polo II
3030 Coimbra (Portugal)
{cfferreira | ivonefamorim | rprocha | jorge}@isr.uc.pt*

***Abstract*—This article reports on a map building method that integrates topological and geometric maps created independently using multiple sensors. The procedure is termed T-SLAM to emphasize the integration of Topological and local Geometric maps that are created using a SLAM algorithm. The topological and metric representations are created independently, being local metric maps associated with topological places and registered at the topological level. The T-SLAM approach is mathematically formulated and applied to the localization problem within the Intelligent Robotic Porter System (IRPS) project, which is aimed at deploying mobile robots in large environments (e.g. airports). Some preliminary experimental results demonstrate the validity of the proposed method.**

***Index Terms* -** Topological Maps, View-based Localization, SLAM Geometric Maps, Robot Localization.

TABLE I: Comparison of Place-recognition successes with and without using HMMs.

K	total number of nodes in the topological map
k	an index, node occupied by the robot or position in the topological map
V_{obs}^{t}	the set of observations at time t
M	total number of distinct metric maps
m	an index, usually employed to denote a particular metric map
$\mathcal{X}_m^t$	representation of the metric map m at time t
α_i	the mixture component coefficient or component prior probability
$\mathcal{V}$	the FIM, or the complete set of views/vectors as collected during Environment-Familiarisation stage
π	the initial probability distribution over the hidden states of the Hidden Markov Model
a_{ij}	the probability of transiting from [hidden] state i to state j in the Hidden Markov Model
$b_i(n)$	the observation or emission probability for the symbol b_i at the place n within the Hidden Markov Model
j	an index, usually employed to denote a particular feature, j
k^*	the estimated place as obtained by applying the maximum criterion to the Belief over the indices of the Reference Sequence
M	the number of [hidden] states in the Hidden Markov Model
N	the number of [visible] observations/symbols in the Hidden Markov Model
Z	Hidden or incomplete data in a Mixture Mode
z_k	the vector from matrix Z corresponding to the view/vector V_k
λ	the parameter set, $\langle N, M, \{\pi_i\}, \{a_{ij}\}, \{b_i(n)\}\rangle$, of the Hidden Markov Model
Θ_i	a single component of the mixture model with the named features

I. Introduction

This article explores the use of combinations of topological and local geometric maps. There are a number of methods in the literature that attempt to exploit the perceived advantages of combined, hybrid or hierarchical maps for use in environment representation and mobile robot localization. There are some methods that are described in the literature that allude to Topological SLAM [1], [2], [3], [4], [5], [6], [7], [8], [9], [10], [11], [12], [13] in order to create an association with (geometric) Simultaneous Localization and Mapping (SLAM), a well-accepted means of building maps [14], [15], [16].

We could classify the methods that utilize both topological and metric information into hierarchical or hybrid methods. While the distinction is primarily semantic in nature, hierarchical methods could be seen as maintaining different representations in order to accomplish different purposes such as long-term planning in the topological map and precise motion control for navigating among obstacles using metric information.

One of the earliest and possibly one of the most well known approaches in this category is the Spatial Semantic Hierarchy or the SSH developed by Ben Kuipers [1], [2]. The SSH is described as a model of knowledge of large-scale space consisting of multiple interacting representations, both qualitative and quantitative. The representation of the environment is maintained in the form of a hierarchy of maps – including metric and topological levels – each of which allows some abstraction of the perception and interaction of the robot with the environment. The advantages gained from using SSH or similar hierarchical model of representations is that incomplete or uncertainty in the information is handled in different forms depending on which particular localization or navigation problem is to be solved. Local metric maps help to perform place recognition, (middle-level) topological maps help create consistent maps in the face of challenges such as loop-closing problems, and the global metric maps maintain an overall consistency in the global position of the robot.

Hybrid approaches are usually employed to resolve specific disadvantages of one representation with regards to the other. In certain approaches that primarily depend on geometric maps, the loop closure problem has been resolved by simultaneously having locally precise geometrical information and globally consistent topological information about a (large) environment.

One such hybrid method is proposed by Choset and Nagatani [3] propose a SLAM method that exploits topology of the free space to localize the robot on a partial map. Low-level control laws are used to generate Voronoi graph (VG) and explore the unknown space. A graph matching process over the VG structure is used for robot localization, whereby the robot locates itself to one of the places of the VG, though the robot does not know its metric coordinates.

Thrun [17] builds a global metric (grid-based) map of the environment and then extracts a topological graph from this metric representation. Besides being not scalable to large environments, this method requires a globally consistent metric map, which is in general very difficult to obtain.

There are also attempts to utilize graph based approaches to solve particular problems that appear at the time of creation of metric maps. Methods such as [5], use graphical methods to maintain hypothesis for map expansion and closure, *i.e.* graph-like methods are used to maintain multiple map hypothesis of the main map which is geometrical. There are works that enhance the applicability of metric maps and the ability of users to interact with these such as representing individual objects. In [18], Limketkai *et al.* store the representation of objects (some of which might also be used by persons) using a technique called Random Markov networks.

Tomatis *et al.* [13] developed a hybrid map representation wherein a global topological map connects local metric maps. The robot may switch between both representation when navigation conditions change (*e.g.* leaving a room and crossing a door). When doing this, the method requires a detectable metric feature in order to determine the transition point where the change from topological to metric has to be done and allows robust initialization of the metric localization (relocation and loop closure). The method was validated within office-like environments but its potential is unclear for different and larger environments.

In [19] Zimmer utilizes a clustering algorithm based on neural networks to cluster the local polar maps and ultimately register them in the global topological map. The experiments were performed on a small environment and the results are unclear regarding the scalability for larger environments. A similar idea is behind the procedure adopted by Zivkovic [20] where panoramic images are grouped and semantic information is associated with the groups. It is stated that this method bears semblance to the way animals represent their environment. As in the case of [4], a clustering approach is used to group images and represent places in the environment by using a typical set of images for that place. In [21], Thomas and Donikian hypothesize a hierarchical set of (topological) representations that represent the environment using similarity of places. The developers of these methods claim that such

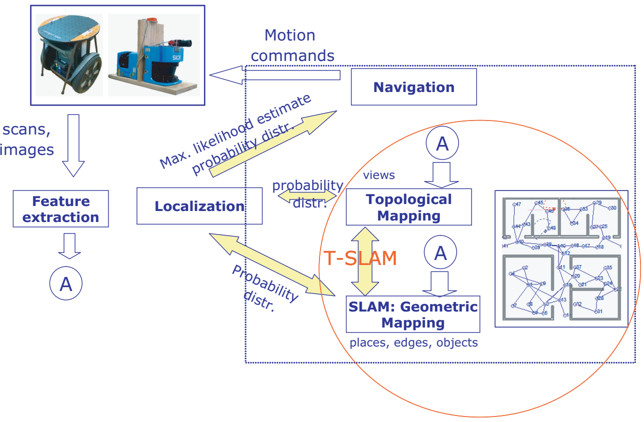

Fig. 1: Depiction of T-SLAM: the use of combined Topological and Geometric maps.

labeling of (similar-looking) places is in line with the spatial concepts that humans employ.

In the current work, we propose a generic method to integrate a global topological map with a set of two or more geometric maps. Some of the nodes of the topological map are associated with the individual metric maps, as depicted in Fig. 1. Our method tracks the global position of the robot only within the topological map. The localisation procedure in the topological map isolates features or properties of the environment into groups that are used to recover the node in the topological map that is currently occupied by the robot. By exploiting the associations between the nodes and the metric maps, we also maintain the local position of the robot enabling the precise geometric positioning of the robot. Localisation in each local metric map is performed independently and simultaneously. Map updating is performed simultaneously in these local metric maps as would be performed in a conventional SLAM algorithm.

In the next section, the method of localisation in the hierarchical representation is presented. In subsection II-B a brief overview of the topological localization method is presented. In section II-C, a selected SLAM algorithm is used to create a geometric map. In section II, the combination of topological and geometric maps is described together with the localization system. In section III, the preliminary result from experiments using combined Topological and geometric maps.

II. Integrating Topological and Geometric Maps

Our representation of the environment is composed of a global topological map and a set of two or more local geometric maps. Let K be the total number of nodes in the topological map, these nodes indexed by the variable $k = 1, \ldots, K$. Let M be the total number of metric maps identified by $m = 1, \ldots, M$. Let $\mathcal{X}_m^t$ denote the representation of the robot in the topological map m at the discrete instance of time t. $\mathcal{X}_m^t$ varies depending on how the position of the robot is maintained in the metric map.

Since there exists a single global topological map and multiple geometric maps conditioned on this global topological

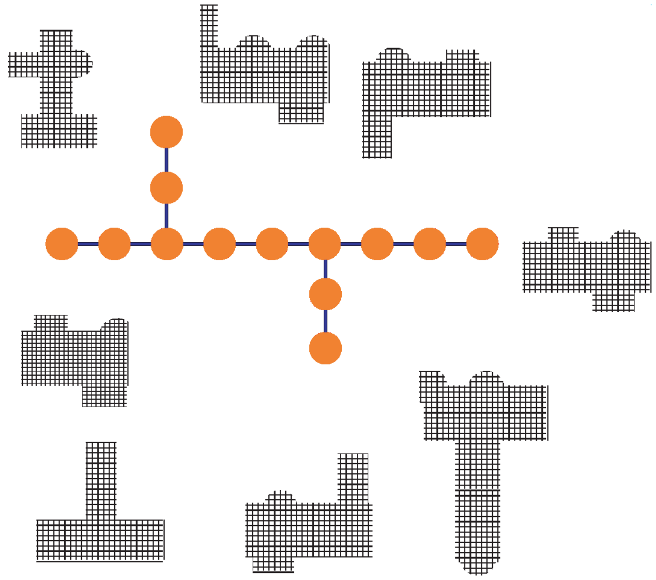

Fig. 2: Depiction of Independent creation of Topological and Metric maps.

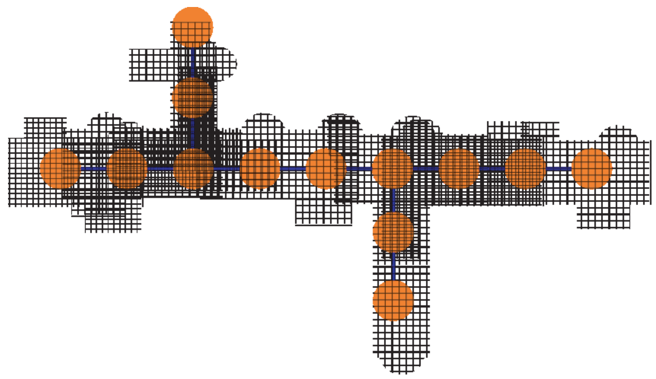

Fig. 3: Depiction of Superimposition that actually exists between the Topological map and the set of geometric maps.

map, as depicted in Fig. 2 and Fig. 3.

The probability of the robot being localised in both, the topological k and metric map m is given by $P(\mathcal{X}_m^t, k|V_{obs})$ in (1).

$$P(\mathcal{X}_m^t, k|V_{obs}) = P(\mathcal{X}_m^t|k, V_{obs}) \times P(k|V_{obs}) \qquad (1)$$

The above expression conditions the probability of localisation on both maps on the probability of localisation on the global topological map. The term $P(k|V_{obs})$ in (1) denotes the localisation in the topological map. Without prejudice to the general case, the index indicating time has been removed from the remaining expressions.

$P(\mathcal{X}_m|k, V_{obs})$ represents the localisation in the metric map, conditioned on the robot being positioned at node k in the topological map and can be expanded as in (4).

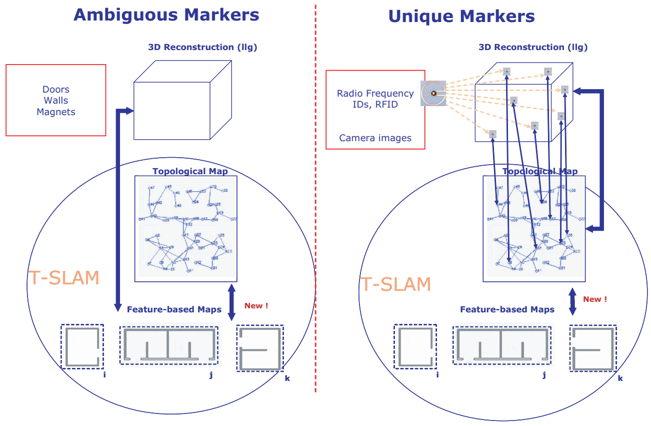

Fig. 4: Registration of Topological and Geometric Maps.

$$P(\mathcal{X}_m|k, V_{obs}) = \frac{P(\mathcal{X}_m, k, V_{obs})}{P(k, V_{obs})} \qquad (2)$$

$$= \frac{P(V_{obs}|k, \mathcal{X}_m) \times P(\mathcal{X}_m|k) \times P(k)}{P(V_{obs}|k) \times P(k)} \qquad (3)$$

$$= \frac{P(V_{obs}|k, \mathcal{X}_m) \times P(\mathcal{X}_m|k)}{P(V_{obs}|k)} \qquad (4)$$

$P(\mathcal{X}_m|k)$ captures the association that the nodes of the topological map have with the individual metric maps. The exact nature of this association can vary depending on the features that are used with the topological and metric map and on the assumptions that are associated with the creation of each type of map. T-SLAM is an attempt to explore one type of association between a set of local geometric maps and a global topological map.

The real advantage of T-SLAM will emerge in scenarios in which the set of metric maps is registered to the global map only at certain places. For example, *some* nodes in the topological map might be associated with *way points* in the metric maps, through the use of artificial environment properties such as Radio Frequency Identification (RFID) tags [22] as depicted in Fig. 4.

The splitting of the environment, into many smaller regions or sections, that is described in this article is not new, see [23], [6], [24], [25] for a recent approach. The advantages of using the approach put forth in this article is that the knowledge of the position of the robot is conditioned on the nodes of the graph, the probability of which is valid globally, over the entire environment i.e. over all the sections of the environment.

A. *Current Problem formulation*

In this article, the first version of T-SLAM is presented. Each node of the topological map is registered with every local geometric map. Each node k, in the topological map is associated with one or more geometric maps m by a human operator. This association is represented in the form of Node-Metric Map association matrix. Each element a_{mk} of this association matrix is assigned the value of 1 is the node is associated with the geometric map, 0 otherwise. Each line in the matrix corresponds to a particular metric map m and each column to a particular node k, leading to the expression (5). The Node-Metric map association matrix allows us to express the probability distribution associated to a map m^*, conditioned on the node k, $P(\mathcal{X}_{m^*}|k)$, by:

$$P(\mathcal{X}_{m^*}|k) = \frac{a_{m^*k}}{\sum_{m=1}^{M} a_{mk}} \quad (5)$$

The global probability of being at a particular position within the set of metric maps m is given by $\sum_{k=1}^{K} P(\mathcal{X}_m^t, k|V_{obs})$ and the location of the robot might be expressed as in (6) where $L(\mathcal{X}_m)$ is the Maximum Likelihood operator. The current observation is used to update the geometric map within which the robot is located.

Additionally, in the current method outlined in this article, we localise the robot in the topological and metric maps independently. This results in the simplification: $P(V_{obs}|k, \mathcal{X}_m) = P(V_{obs}|k) * P(V_{obs}|\mathcal{X}_m)$.

$$L(\mathcal{X}_m) = \mathcal{MLE}_k(\sum_{k=1}^{K} P(V_{obs}|\mathcal{X}_m) \times \frac{P(k|V_{obs})}{\sum_{m=1}^{M} a_{mk}}) \quad (6)$$

In the following subsection, II-B, an expression will be developed for $P(k|V_{obs})$, where the topological map is built from a sequence of raw-image sequences. In section II-C, a well-known SLAM algorithm is utilised to create metric maps and localise the robot within them.

B. Topological Maps from Raw Sensor Data

In [26], a procedure was developed to localize a robot as it travelled along a path. During a first trip around the environment, the Environment Familiarization phase, depicted at left in Fig. 5, the robot samples the environment according to a sampling plan, collecting features by using its various sensors into the Reference Sequence.

A repetition of the motion performed during the place recognition should propel the robot along the path described by the Reference Sequence. Any maneuver other than the ones taken during the Environment Familiarization phase will take the robot to a place that was not sampled in the Environment Familiarization phase. The $Lost\_Places$, in all a total of K in number, accommodate these possible views. Thus, each $Lost\_Place$ takes into account the fact that the robot might be seeing views that were not seen in the Environment Familiarization phase.

The sampled views, which would normally be modelled as a left-to-right graph as at right in Fig. 5, are augmented by the insertion of '$Lost\_Places$' as depicted in Fig. 6.

The sequence begins with P_{Lost_0} which indicates that the robot is completely lost or has never localized. Also, before every original place P_i, there is a P_{Lost_i}. By moving forward from one $Lost\_Place$, the robot can transition from P_{Lost_i} to any node P_k where $k > i$. Similarly, from P_i the robot can transition to $P_k : k > i$ or to P_{Lost_i+1}. The graph does not allow a single-step transition from one P_{Lost_i} to another P_{Lost_j}.

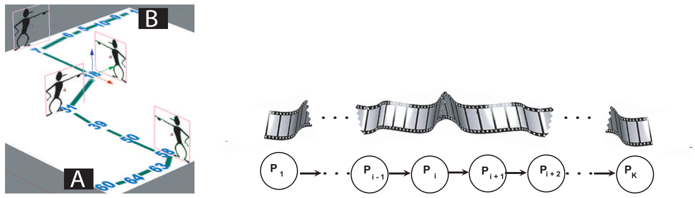

Fig. 5: The robot is led through the environment on the Environment Familiarization run to create the Reference Sequence (left). This Reference Sequence constitutes a left-to-right graph (right) composed of 'K' views, ordered as they were sampled during the Environment Familiarization.

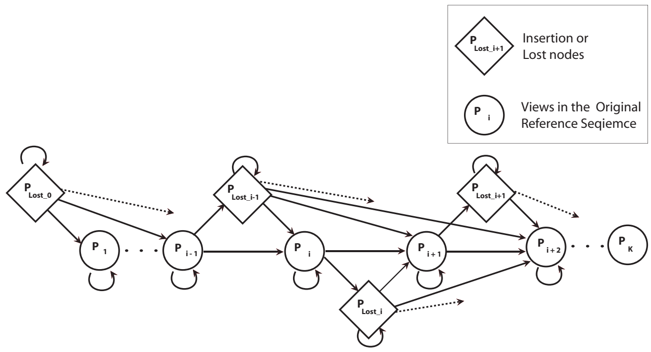

Fig. 6: The figure depicts a modified Markov chain, with '$Lost\_Places$' inserted within the original Reference Sequence, to perform place-recognition. The dotted lines indicate the transitions to the Places in the original Reference Sequence and which have not been drawn to avoid cluttering the figure.

When the robot needs to localize itself, it moves through the same environment, the current view is compared to the previously collected views and an inference is made of the current position of the robot. A Hidden Markov Model (HMM) is used to perform place recognition using the modified Markov Chain in Fig. 6 as a model for the transitions between the hidden states of the HMM. The Viterbi algorithm is commonly used in the context of HMMs to determine the most probable sequence of hidden states that gave rise to a particular sequence of observations. It is an inference tool that is associated with the process of making inferences in a HMM and is utilized to position the robot within the Reference Sequence by using the current sequence of observations. The HMM is specified in terms of its parameters λ, as in (7), where N corresponds to the number of states in the HMM, M the number of observations that will be used to make the inference, π represents the initial probability on the states, the a_{ij}s correspond to the transition probabilities between a pair of states i and j and $b_i(n)$ represents the probability of viewing symbol n at state i.

$$\lambda = \langle N, M, \{\pi_i\}, \{a_{ij}\}, \{b_i(n)\}\rangle \quad (7)$$

The transition between the states is influenced by the transition probabilities between a pair of places in the graph shown in Fig. 6. An elementary robot motion model is developed to evaluate the transition probability matrix. For each sequence of M observations, a simple distribution is used to model the transition probability distribution from each $Lost\_Place$ to the remaining original places in the Reference Sequence favoring places that lie closer in the Reference Sequence. The transition probability leading away from any of the original places in the Reference Sequence is uniformly split between the next original place (to the right) and to the corresponding $Lost\_Place$. The one-step transition probability from one $Lost\_Place$ to another $Lost\_Place$ is zero.

The first hidden state is always matched to the first $Lost\_Place$, P_{Lost_0}. This P_{Lost_0}, has a non-zero probability of reaching any place in the original Reference Sequence.

The observation model of the HMM is based on matching the current view with the views in the Reference Sequence. In the absence of any information regarding the view that will be visible at the '$Lost\_Place$', we arbitrarily define the observation probability as an Uniform distribution over the K view in the original Reference Sequence. The features from each view in the Reference Sequence are converted into binary form and are represented within a Feature Incidence Matrix (FIM), $\mathcal{V}$. Due to the large dimensionality of the FIM, it is subsequently modelled as a Bernoulli Mixture Model (BMM). The parameters of the BMM are obtained by running the Expectation Maximization(EM) algorithm.

The Mixture parameters and the posterior probabilities over the components, the Z terms in (8), are used to evaluate the likelihood as depicted in (8), $P(V_k)$ representing the prior probabilities over each view k, in the Reference Sequence. As expressed in (9), the *Maximum Likelihood Estimation* is used to obtain the index k^* in $\mathcal{V}$ that best describes the object to be matched, V_{obs}.

$$P(k|V_{obs}) = \frac{\sum_{j=1}^{M} P(V_k) z_{kj} \alpha_j P(V_{obs}|\Theta_j)}{\sum_{k=1}^{K} \sum_{j=1}^{M} P(V_k) z_{kj} \alpha_j P(V_{obs}|\Theta_j)} \tag{8}$$

$$P(k^*|V_{obs}) = \underset{k}{\overset{K}{\arg\max}}\, P(k|V_{obs}) \tag{9}$$

C. Creating and Updating local Metric Maps using SLAM

The incremental creation of Geometric maps from sensor data has been an area of much research over the last two decades. Simultaneous Localization and Mapping SLAM and Concurrent Mapping and Localization, CML, algorithms have been proposed by various researchers for the creation of different geometric maps. These algorithms have been very successful in the creation and utilisation of maps in indoor environments [14].

A couple of state of the art SLAM algorithms was used to create the local geometric maps. We experimented with the DP-SLAM [15] and the Fast-SLAM [27] algorithms. Both methods create grid-based metric maps using particle filters. The local geometric maps presented in this article were created using the Fast-SLAM algorithm.

A particle filter is a method of obtaining a description of a certain state space through partial observations of that space, which inevitably contain measurement errors. It maintains a weighted, and normalized, set of sampled states, $S = \{s_1, s_2, \ldots, s_p\}$, called particles. At each step, and given an observation vector z and a control vector u (in our context), the particle filter:

1) Samples m new states $S' = \{s'_1, s'_2, \ldots, s'_p\}$ from S, with replacement, using the probability density given by the weights of the elements in S.
2) Runs the state given by each particle through the corresponding motion model, using the previous states and u, obtaining in this way the new generation of particles.
3) Each new particle is then weighted, using the observation model together with the vector z.
4) Normalizes the weights of the new set of states.

The Fast-SLAM algorithm [27] is known for the speed at which the map is updated and for the relatively good quality of the geometric maps that are outputted. While the original Fast-SLAM algorithm [27] procedure was developed for metric maps using landmark, modifications and improvements were subsequently made including an adaptation to grid-based maps [16]. An implementation of this algorithm was obtained from the Open-SLAM web page [28]. In our current work we have create adopted a gird-based

III. Experiments and Results

Initial experiments have been performed on the localisation using a global topological map and a set of multiple metric maps. The topological representation of the environment was maintained in the form of a sequence of laser range scans and images gathered while leading the robot along one or more paths in the environment.

Our robot platform is equipped with two cameras and a Laser Range Finder, LRF, as seen in Fig. 7. The acquisition of data from the sensors and the control of the robot is performed within CARMEN. The two cameras, one facing forwards and the other facing onto one side, are capable of taking gray-scale 640x480 images. SIFT features [29] are utilised to perform matching between current observations and previously obtained images.

The forward-facing LRF provides a set of 361 range measurements through a 180 degree interval. Features from this sensor are used within the topological representation of the environment. The raw data from the sensor is used directly by the SLAM algorithm to build and maintain the topological map

The robot was first led along a path, depicted in Fig. 8, to create the topological and the set of geometric maps. The images from the cameras and the LRF were used to create the topological representation of the path, while raw laser range finder data and odometry were used to create the geometric maps. A new geometric maps was created after a specific

Fig. 7: The sensor platform comprising of two laser range finders and two cameras is mounted on the Segway RMP 200.

TABLE II: The Node-Metric Map Association Matrix for experiment 1.

	1	2	...	53	54	55	...	94	95	96	...	143	144
m=1	1	1	...	1	0	0	...	0	0	0	...	0	0
m=2	0	0	...	1	1	1	...	1	0	0	...	0	0
m=3	0	0	...	0	0	0	...	1	1	1	...	1	1

amount of time of robot travel. A set of three geometric maps were created in all as shown in Fig. 9.

As stated in section II-A, the association between places that are represented in the topological map and the individual metric maps is represented in a Node-Metric maps association table. Excerpts of this map are shown in Table II.

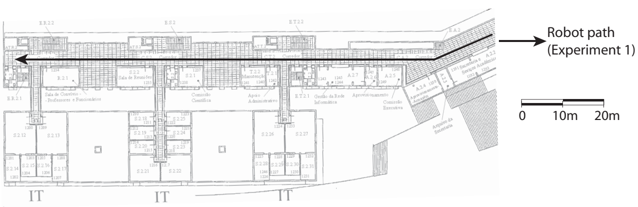

Fig. 8: Experiment 1: The robot was driven along a long hallway and map-building and localisation were performed to create independent topological and geometric representations.

In a second experiment, the robot was driven along a path lined primarily by glass panes and pillars, Fig. 10. Typically, such an environment is difficult for SLAM applications given the absence of features in the direction lateral to the direction of robot travel. Excerpts of the Node-Metric Map association matrix are shown in Table III. A few images from the set of 150 images that were used to construct the topological representation are presented in Fig. 11. As is seen in the above image, this environment, the robot is often surrounded by reflective and glazed surfaces, which make the SLAM difficult. The combined maps are depicted in Fig. 12.

There is some super position since the individual paths are created incrementally. Some of the larger amount of overlap that is present between the sections is removed during the process of merging topological paths. A small amount of overlap is maintained to allow transition between paths.

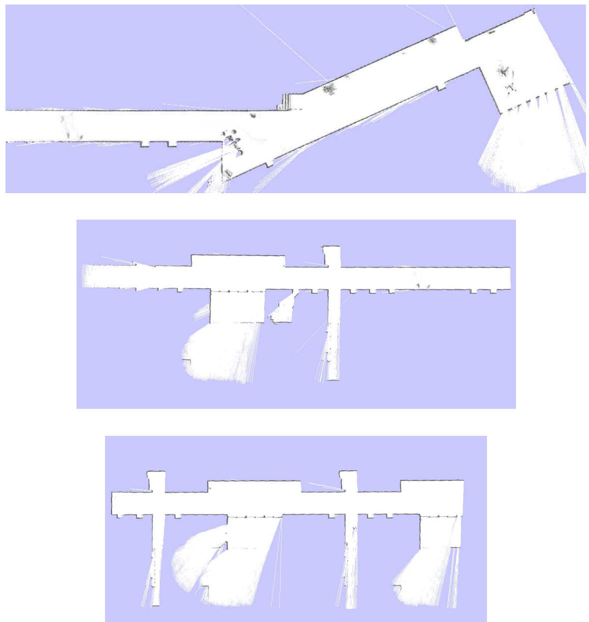

Fig. 9: Experiment 1: The set of three metric maps that are created by running the Fast-SLAM algorithm after the initial run through the environment in experiment 1.

TABLE III: The Node-Metric Map Association Matrix for experiment 2.

	1	2	...	54	55	55	...	114	115	116	...	149	150
m=1	1	1	...	1	0	0	...	0	0	0	...	0	0
m=2	0	0	...	1	1	1	...	1	0	0	...	0	0
m=3	0	0	...	0	0	0	...	1	1	1	...	1	1

IV. Conclusions

Initial results were presented in this work on the simultaneous use of one global topological whose nodes are registered with multiple metric maps.

Current work includes the improved registration of the topological map with each metric map such that the uncertainty in the topological map can be transferred over to the metric maps and vice versa. We expect that this will lead to increased robustness in the localisation within the geometric maps and to reliable loop closing procedures in the topological map.

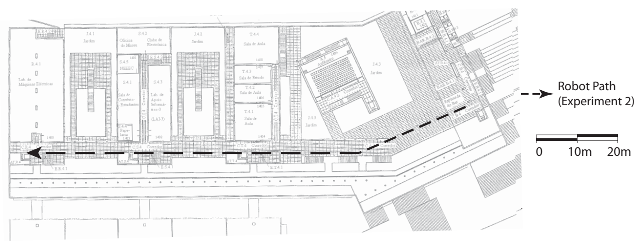

Fig. 10: Experiment 2: Mapping and localisation is performed in a second environment that comprises pillars and glass surfaces.

Fig. 11: Experiment 2: Typical images from a set of 150 images that comprise the topological representation of the path.

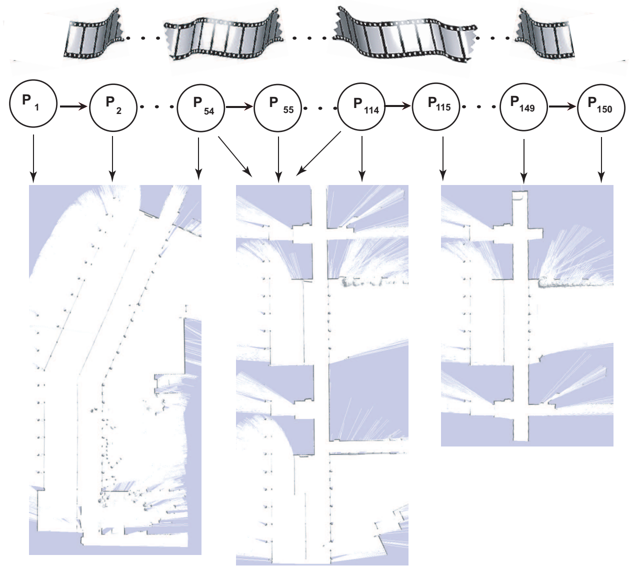

Fig. 12: Experiment 2: The association of the nodes of the topological map with the set of three metric maps. In the current version of T-SLAM, the registration of topological and metric maps is maintained in the form of the Node-Metric Map association matrix.

REFERENCES

[1] B. J. Kuipers. The Spatial Semantic Hierarchy. *Artificial Intelligence, Elsevier*, 119:191–233, 2000. http://www.cs.utexas.edu/users/qr/papers/Kuipers-aij-00.html.

[2] Benjamin Kuipers, Joseph Modayil, Patrick Beeson, Matt MacMahon, and Francesco Savelli. Local Metrical and Global Topological Maps in the Hybrid Spatial Semantic Hierarchy. In *IEEE International Conference on Robotics and Automation (ICRA 2004).*, 2004.

[3] H. Choset and K. Nagatani. Topological simultaneous localization and mapping (SLAM): toward exact localization without explicit localization. *IEEE Trans. on Robotics and Automation*, 17(2):125–137, 2001.

[4] I. Posner, D. Schroeter, and P. Newman. Using scene similarity for place labelling. In *10th International Symposium on Experimental Robotics, ISER 2006*, 2006.

[5] John Folkesson and Henrik Christensen. Graphical slam - a self-correcting map. In *IEEE International Conference on Robotics and Automation, 2004*, 2004.

[6] Brad Lisien, Deryck Morales, David Silver, George Kantor, Ioannis Rekleitis, and Howie Choset. Hierarchical Simultaneous Localization and Mapping. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robotic Systems, IROS*, pages 448–453, las Vegas, Nevada, October 2003.

[7] Agostino Martinelli, Viet Nguyen, Nicola Tomatis, and Roland Siegwart. A relative map approach to slam based on shift and rotation invariants. *Robotics and Autonomous Systems*, 55(1):50–61, January 2006.

[8] P. Newman, D. Cole, and K. Ho. Outdoor slam using visual appearance and laser ranging. In *IEEE International Conference on Robotics and Automation, 2006*, 2006.

[9] Max Pfingsthorn, Bayu Slamet, and Arnoud Visser. A scalable hybrid multi-robot slam method for highly detailed maps. In *Proceedings of the 11th RoboCup International Symposium*, July 2007.

[10] A. Arleo, J. del R. Millán, and D. Floreano. Efficient learning of variable-resolution cognitive maps for autonomous indoor navigation. *IEEE Trans. on Robotics and Automation*, 15(6):990–1000, Dec. 1999.

[11] J. Gasós and A Saffiotti. Integrating fuzzy geometric maps and topological maps for robot navigation. In *Proc. of 3rd Int. ICSC Symp. on Soft Computing (SOCO'99)*, pages 754–760, Genova, Italy, Jun. 1999.

[12] E. Aguirre and A. González. Integrating fuzzy topological maps and fuzzy geometric maps for behavior-based robots. *Int. Journal of Intelligent Systems*, 17:333–368, 2002.

[13] N. Tomatis, I. Nourbakhsh, and R. Siegwart. Hybrid simultaneous localization and map building: a natural integration of topological and metric. *Robotics and Autonomous Systems*, 44:3–14, 2003.

[14] S. Thrun. Robotic mapping: A survey. In G. Lakemeyer and B. Nebel, editors, *Exploring Artificial Intelligence in the New Millenium*. Morgan Kaufmann, 2002.

[15] A. Eliazar and R. Parr. Dp-slam: Fast, robust simultaneous localization and mapping without predetermined landmarks. In *Proc. 18th Int. Joint Conf. on Artificial Intelligence (IJCAI-03)*, pages 1135–1142, 2003.

[16] C. Stachniss, D. Haehnel, W. Burgard, and G. Grisetti. On actively closing loops in grid-based fastslam. *The International Journal of the Robotics Society of Japan (RSJ)*, 19(10):1059–1080, 2005.

[17] Sebastian Thrun. Learning metric-topological maps for indoor mobile robot localization. *Artificial Intelligence*, 99(1):21–71, 1998.

[18] Benson Limketkai, Lin Liao, and Dieter Fox. Relational object maps for mobile robots. In *International Joint Conference on Artificial Intelligence (IJCAI)*, 2005.

[19] U. R. Zimmer. Embedding local metrical map patches in a globally consistent topological map. In *Symposium on Underwater Technology 2000, Tokyo, Japan, May 23-26 2000*, May 2000.

[20] Zoran Zivkovic, Olaf Booij, and Ben Kröse. From images to rooms. *Robotics and Autonomous Systems*, 55(5):411–418, 2007.

[21] Romain Thomas and Stéphane Donikian. A model of hierarchical cognitive map and human memory designed for reactive and planned navigation. Technical report, l'Institut National de Recherche en Informatique et en Automatique, INRIA, Project siames, 2000.

[22] D. Haehnel, W. Burgard, D. Fox, K. Fishkin, and M. Philipose. Mapping and localization with rfid technology. In *IEEE International Conference on Robotics and Automation (ICRA), 2004*, 2004.

[23] Jose-Luis Blanco, Juan-Antonio Fernández-Madrigal, and Javier González. Toward a unified bayesian approach to hybrid metric–topological slam. *IEEE Transactions on Robotics*, 24(2):259–270, 2008.

[24] Khashayar Rohanimanesh, Georgios Theocharous, and Sridhar Mahadevan. Hierarchical map learning for robot navigation. In *In AIPS Workshop on Decision-Theoretic Planning*, 2000.

[25] Georgios Theocharous, Kevin Murphy, and Leslie Pack Kaelbling. Representing hierarchical pomdps as dbns for multi-scale robot localization. In *IEEE International Conference on Robotics and Automation, 2004*, 2004.

[26] F. Ferreira, V. Santos, and J. Dias. A topological path layout for autonomous navigation of multi-sensor robots. *International Journal of Factory Automation, Robotics and Soft Computing*, 1:203–215, 2007.

[27] M. Montemerlo, S. Thrun, D. Koller, and B. Wegbreit. FastSLAM: A factored solution to the simultaneous localization and mapping problem. In *Proceedings of the AAAI National Conference on Artificial Intelligence*, Edmonton, Canada, 2002. AAAI.

[28] www.openslam.org.

[29] David G. Lowe. Distinctive Image Features-From Scale-Invariant Keypoints. *IJCV*, 60, Issue 2(Issue 2):91–119, November 2004.