

A Markov semi-supervised clustering approach and its application in topological map extraction

Ming Liu, Francis Colas, François Pomerleau, Roland Siegwart
Autonomous Systems Lab, ETH Zurich, Switzerland
[firstname.lastname@mavt.ethz.ch, rsiegwart@ethz.ch]

Abstract—In this paper, we present a novel semi-supervised clustering approach based on Markov process. It deals with data which include abundant local constraints. We apply the designed model to a topological region extraction problem, where topological segmentation is constructed based on sparse human inputs (potentially provided by human experts). The model considers human indications as seeds for topological regions, i.e. the partially labeled data. It results in a regional topological segmentation of connected free space.

I. INTRODUCTION

For the past decade, more and more robotic systems are deployed for search and rescue missions [8], [25]. It is not only because robots could work in extreme cases where not feasible for human existence, also because they can build detailed models of target environments, which provide valuable references for rescue missions. Though the state-of-art robotic technology has made a lot of progresses in the autonomy for search and rescue missions [20], [25], the co-operation between human rescuers and robots is still required, since it is an important way to leverage the expertise of both human and the robot. As the basis of such cooperation, the pattern that how human expert communicate with robots and share a common understanding of the environment is essential. Burke et al [8] identified intention recognition as one of the most important technological challenges in this research field.

A. Urban Search and Rescue

For search and rescue missions, the aim is to have robots and human working together as a multi-agent team. However, not much work explored the utilization of human experience in mapping process [31], [24]. As a major task for robots, building a topological environment model of the scene is the basis for the cooperation among team members, while humans have a mostly topological representation of their environments [23]. On the other hand, the output information from state-of-art robotics mapping methods [12], [30] is dense and mostly based on metric measurements. There is a missing link between the raw metric map and the way humans usually use to understand the world. Therefore, several topological mapping techniques [21], [6], [11] have been developed to bridge this gap. Most existing approaches try to extract the topological properties of a given environment automatically without user input. In our previous report [15],

we showed how can such a regional topological map be integrated in a multi-agent team, aiding the navigation and mission planning.

Since the topological map is solely created by the robot, usually it doesn't allow human users to flexibly manipulate its structure. A comprehensive consistent understanding of the environment is usually not available in such human-robot multi-agent system. For real applications in search and rescue which involve human-robot cooperation, it is an inneglectable shortcoming. Especially, as human experts would rather follow their routines in dealing with emergency cases [1] considering their own experience, the autonomy of the robot can be possibly ignored by the operators [19]. Therefore, the fusion of human expertise is a key to efficiently develop the autonomy of the robot in a human-robot team.

However, it is not easy to directly use the input hints from human in the existing topological mapping methods [21], [6]. The most important issue is - indications from human are sparse. Furthermore, this subject is not merely about human-robot interfaces or robot mapping in rescue missions [2] but also how to unify the understanding of the environment for both human and robots. For example, in a car accident scenario, the commander would rather just point to one *single* point in the map, stating that "There may be survivors in this *region*, behind the crashed red car. Send your men there to have a better check." Human can interpret the regional information quite well. But this indication assumes too much context and inference for a robotic system to disambiguate. In this paper, we propose a concise semi-supervised clustering method based on probability theory and Markov process to tackle this problem, i.e. how to change the sparse indication to the latently referred meaning by generating a regional topological map especially for semi-structured environments.

B. Topological map for semi-structured environments

The problem that we try to tackle is closely related to topological mapping based on distinct regions. So far, most works in that direction aim at segmentation of structured environments, using spectral clustering [10], trumped Generalized Voronoi Graph (GVG) [28], graph based relaxation [13] or other Bayesian based methods [7]. However, little progress has been made in the segmentation problem for semi-structured environments. The term semi-structured environment intuitively means that the environment can not be partitioned in sense of structures such as "rooms," "corridors"

etc, whereas these structures could be easily identified for indoor environments. Therefore, it is important to define local structures, by which the local homology of the segmented regions can be maintained. As one of the typical data structures [29], grid-map hinders the performance of topological segmentation algorithms in the following aspects.

- *Abundant local constraints.* Salient corners and ill-shaped obstacles can be constantly observed. It is hard for a global clustering algorithm to consider these local constraints. For example, the different sizes of free regions could cause scaling problems [26] for spectral sensitive methods, such as spectral clustering [10].
- *Highly noisy observation.* *Ghost obstacles* may be observed during mapping process due to moving objects or reflecting surfaces. The noisy data with abundant local constraints make it hard for existing clustering methods [3], [14] to converge to global minimums.

These issues show that a clustering method based on minimization of global cost function may not be feasible for this problem. Inspired by this deduction, we present an iterative semi-supervised clustering method, which takes locally embedded relations between neighbor data into account. It shows more rational segmentation results than existing approaches.

C. Semi-supervised clustering

The clustering problem stated in this paper is closely related to the field *semi-supervised clustering* [16]. It has been a very active field during the past decade. The existing algorithms can be categorized as two main directions: first, most work focused on adapting unsupervised clustering, such as K-means [3], [4] or kernel K-means [18]; others mainly worked on heuristic algorithms. For example, [32] was based on factorization of similarity matrix; [22] took a Maximize-a-posterior (MAP) result of a Gaussian Process Classification etc. The nature of the grid-map data indicates that methods using global optimization[5], [9] can hardly model the abundant local constraints efficiently. Therefore, a semi-supervised clustering method that takes local constraints into account is required.

Sparse supervision is considered as the initial estimation of an inherent topological map implied by the supervision, namely “seeds”[3]. We assume that these labeled data provide the number of nodes and coarse positions of the nodes for the final clustering results. Then each cluster will expand its territory iteratively by sampling the labels of its k -nearest neighbors. It can be seen that the current state of clustering is only related to the previous iteration, which models a Markov process which *flooding fill* in all neighboring directions.

Apart from this, this paper is organized as follows. In the next section, a novel semi-supervised clustering algorithm will be presented and compared with other related methods on generic test data. The application of the algorithm in topological mapping will be introduced in section III. The model update and label update for topological mapping are presented in sections IV and V respectively. We stress the difference between our experiment results and another

traditional clustering method in section VI, followed by conclusion and outlook of our future work.

II. SEMI-SUPERVISED CLUSTERING USING A MARKOV PROCESS

Semi-supervised clustering is to use a small number of labeled data to aid the clustering of unlabeled data. We propose a novel iterative algorithm to realize semi-supervised clustering considering local constraints. It considers a labeling problem as a Markov process, where each intermediates state stands for a distribution of labels over data points. The goal is to preserve the locality, namely, local constraints as much as possible in the final clusters. Topologically speaking, the clustering process creates a projection, which brings a certain datapoint from the configuration space to a graph space. It has been proved that the local embedding relations to k nearest neighbors are preserved for such projection [27]. So we use a voting process of k nearest neighbors for certain candidate, which is summarized in Algorithm 1.

A. Algorithm

Algorithm 1: Proposed Semi-supervised clustering

Input:
Set of data points $X \leftarrow \{x_1, x_2, \dots, x_N\}, x_i \in \mathbb{R}^d$,
number of clusters C ,
initially labeled data set $S = \{s_1, \dots, s_c, \dots, s_C\}$,
models for each cluster M_c .

Output:
A *partitional clustering* [16] result $\{X_1, \dots, X_c\}$, where $\bigcup_c X_c = X$;

- 1 **while** $\#(\text{unclustered points}) > 0$ **and** *!stop* **do**
- 2 Denote the set of *labeled data*: \hat{X} ;
- 3 Get set of nearest neighbors $\Lambda \subset X \setminus \{\hat{X}\}$ of \hat{X} ;
- 4 **for** *each unlabeled point* $x_u \in \Lambda$ **do**
- 5 $Knn_u \leftarrow k\text{-nearest-neighbor}(x_u, k, X)$;
- 6 each labeled point in Knn_u votes for the cluster
 of x_u using its own label;
- 7 update the set for *labeled data* \hat{X} ;

The model for each cluster M_c is a topology which should at minimum including the following properties:

- 1) The distances definition between any pair of points
- 2) The number of nearest neighbors k that should be considered for $k\text{-nearest-neighbor}()$. The selection of k depends on the expected sparseness of the target dataset.
- 3) A stop condition for the cluster. E.g. when the distance to the nearest unlabeled point is greater than a rational threshold, the cluster model should be considered as stable, hence the clustering should not extend the current cluster further.

The clustering process results in the following features

- The input data will be partitioned into k clusters;

- By denoting the sum of the distances of k -nearest neighbors to x_i as Ψ_i^{knn} , the average of Ψ_i^{knn} in each cluster is minimized.

B. Validation

We evaluated the proposed algorithm on a commonly used 2-ring dataset as shown in figure 1. We do not consider a kernel-based projection of the data space in these results, so that all the algorithms in comparison use Euclidean distance as local constraints. Figure 1(a) depicts the raw data and two labeled datapoints as supervision. The labeled data are marked in red and blue dots. The proposed method is compared with two typical algorithms as follows. Figure 1(b) shows the seeded-Kmeans [3], which is the most widely cited semi-supervised variation for standard K-means. The two curves in the figure show the trajectories of the iterated mean positions for both clusters respectively. Because of the limitation that seeded-Kmeans needs to find the global optimal of the objective function in Euclidean space, the two rings can not be correctly classified. Figure 1(c) shows the result using affinity-propagation [14], which is a typical iterative unsupervised clustering method using local message passing. We could see that although the clusters are locally well-defined, the labeled data are omitted and there are more than two clusters in the final classification result. In figure 1(d), we show the result using the proposed algorithm¹. The linking arrows among datapoints indicate the process of how the points are treated in sequence. We can see that the two rings can be correctly clustered. Please notice that we do not use explicit pairwise constraints, such as *must-link* or *cannot-link*. Intuitively, these constraints are embedded in k -nearest neighbors.

C. Reasoning

As for iterative approaches, the results in figure 1(b) and (c) can be considered as two extreme types of clustering. Seeded-Kmeans considers only global constraints of the data; in the contrary, affinity propagation maximize the importance of local constraints. Nevertheless, the proposed method is compromised. On one hand, it globally considers the clustering problem as a complete process, hence the labeling is updated upon global distribution of the data points. On the other hand, the evolution of each cluster is entirely based on the local voting results, which takes the common opinion of k nearest neighbors for each unlabeled data point. As a result, the global constraints and local constraints are merged efficiently.

III. PROBLEM FORMULATION FOR SEMI-SUPERVISED TOPOLOGICAL SEGMENTATION

In this section, we introduce how could the proposed model convert sparse information to regional definitions for robot mapping. We firstly uniformly decompose the free space of

¹Empirically we choose k between 5~15, which will lead to the same result. Spectral clustering will show a similar result [17]. But since it is not an iterative method, it is not discussed here. Further explanation of how can spectral clustering be integrated and especially analysis of its shortcomings are discussed in our previous work [21].

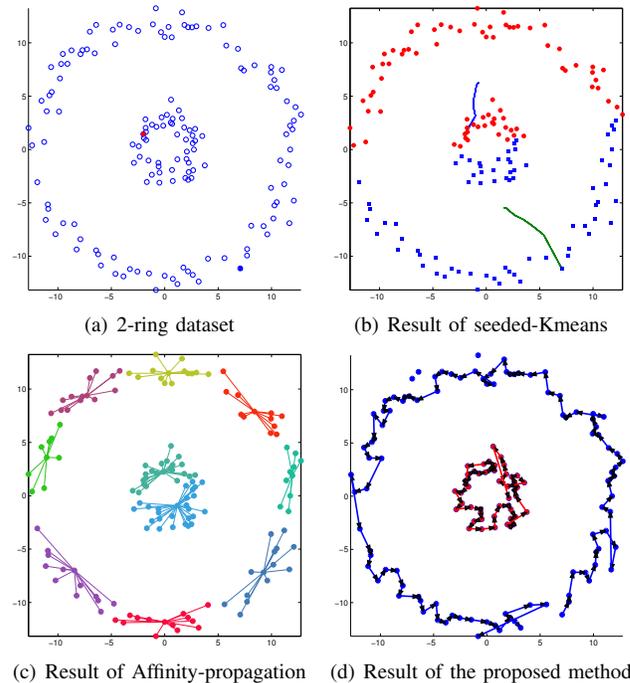


Fig. 1. Better see in color. Comparison of results on a typical 2-ring dataset

the metric map using small grid cells, as shown in figure 2, in order to relief the computational complexity.

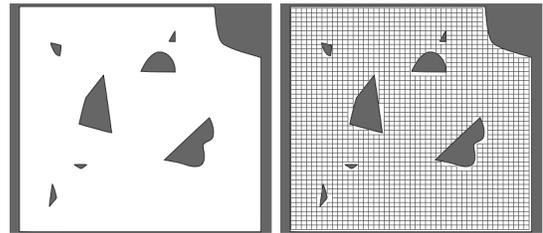


Fig. 2. Raw map and decomposition results

After that, the topological problem is considered as a labeling problem for each grid cell in the map. As a convention, each region in the topological map is considered as a node. We represent each node of the topological map by a model M_i , which contains properties of the node such as the mean position μ_i , and the maturity of the representation ξ_i (stop condition for the model) etc.² Therefore, to obtain a topological map, we need to label all the cells according to local isomorphism over cell distributions.

A. Definitions

The random variables that we consider in this model are:

- *Position* μ_i is the gravity center of topological region i in the metric map;
- *Maturity* ξ_i is a measure of the potential that region i will keep growing. When maturity ξ_i is bigger, it indicates that the region i is less likely to keep growing. We use this parameter to represent a gentle stop condition for the cluster. It controls the growing speed of a region.;
- *Model* M_i of a region combines μ_i and ξ_i ;

²Further properties such as convexity may be considered, by following the same scheme.

- *Label* L_n contains the information that how all the cells are labeled. n is the index of a cell on the free space of the known metric map.

B. Model

Starting with the human indications, the final topological segmentation is considered as the final state of a Markov process. We depict an updating law of the state estimation as follows, assuming Markov property as the proposed algorithm:

$$\begin{aligned} & P({}^t\mathbf{L}_N, {}^t\mathbf{M} \mid {}^{t-1}\mathbf{L}_N, {}^{t-1}\mathbf{M}) \\ &= P({}^t\mathbf{L}_N, {}^t\mathbf{M} \mid {}^{t-1}\mathbf{L}_N, {}^{t-1}\mathbf{M}, {}^{t-2}\mathbf{L}_N, {}^{t-2}\mathbf{M}, \dots) \end{aligned} \quad (1)$$

where ${}^t\mathbf{L}_N$ is the distribution of labels over the N discretized cells, at iteration t . N is the number of free cells in the map. It indicates that the labeling at the current iteration is only related to the latest previous states. $\mathbf{M} : \{M_1, M_2, \dots, M_K\}$, where K is the number of nodes, doesn't depend on the previous labeling. Throughout all this article, bold font is used to represent vectors.

Since the model of a node is not related to whatever label given to it, this updating law can be further decomposed into the model update and label update phases. The update can be factorized as follows.

$$\begin{aligned} & P({}^t\mathbf{L}_N, {}^t\mathbf{M} \mid {}^{t-1}\mathbf{L}_N, {}^{t-1}\mathbf{M}) \\ &= \left[\prod_{i=1}^K P({}^tM_i \mid {}^{t-1}M_i) \right] P({}^t\mathbf{L}_N \mid {}^{t-1}\mathbf{L}_N, {}^t\mathbf{M}) \end{aligned} \quad (2)$$

It shows the estimation of the current states at iteration t given the previous ones. Here we assume the models of different nodes are independent.

Hence if we use hatted parameters to represent the estimation. The graph representation of the update law is shown as figure 3. The model of regions M_i has been expanded to its properties μ_i and ξ_i .

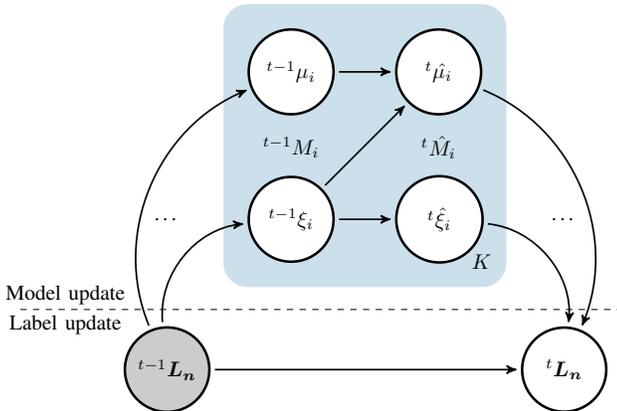


Fig. 3. Graph Model

As depicted in figure 3, ${}^{t-1}\mathbf{L}_n$ is the only directly observable variable from the previous step $t - 1$. In model update phase, the topological model ${}^{t-1}M_i : \{{}^{t-1}\mu_i, {}^{t-1}\xi_i\}$ is inferred from the labeling status. ${}^{t-1}M_i$ provides reference for estimating new position and maturity. According

to algorithm 1, each cluster first defines which unlabeled neighbors need to be considered as candidates for labeling. Then newly inferred labeling ${}^t\mathbf{L}_n$ is achieved by label update phase, namely by local voting results. Model update and label update are introduced in the next two sections respectively.

IV. MODEL UPDATE

In this section, we introduce how the model is updated, i.e. how do μ and ξ evolve over iterations. As defined previously, the model update is represented as:

$$P({}^tM_i \mid {}^{t-1}M_i) = P({}^t\mu_i, {}^t\xi_i \mid {}^{t-1}\mu_i, {}^{t-1}\xi_i) \quad (3)$$

The position of the node μ_i and its maturity ξ_i are independent, on the condition of knowing the labeling status, as ${}^{t-1}\mathbf{L}_n$ d-separate them in the graph. Besides, μ_i at iteration t doesn't depend on the labeling at $t - 1$. The model update can therefore be factorized as:

$$P({}^tM_i \mid {}^{t-1}M_i) = P({}^t\mu_i \mid {}^{t-1}\mu_i, {}^{t-1}\xi_i)P({}^t\xi_i \mid {}^{t-1}\xi_i) \quad (4)$$

Following equation 4, we consider position update and maturity separately. The model update defines which cells are to be labeled in this iteration, reflecting step 3 in algorithm 1.

A. Position update

The first part of equation 4 shows that the change of the mean position of a node is related to the maturity of the region. Intuitively, the more mature a region is, the less probable the position will change. The distribution over the next position is therefore specified as a Gaussian distribution centered on the previous position ${}^{t-1}\mu$, whose variance is related to the maturity, as follows:

$$P({}^t\mu_i \mid {}^{t-1}\mu_i, {}^{t-1}\xi_i) \sim N({}^t\mu_i; {}^{t-1}\mu_i, \sigma_i) \quad (5)$$

where $\sigma_i = \nu e^{\lambda \cdot {}^{t-1}(1-\xi_i)}$. ν is a parameter that defines the maximum step of the position change at each iteration, and λ a parameter that defines the strength that maturity affects the position change. It shows that when the region tend to be mature, hence $\xi_i \rightarrow 1$, the position of the region stays unchanged,

B. Maturity update

Maturity ${}^t\xi_i$ represents the following ratio in practice.

$${}^t\xi_i = \frac{\#\text{stable cells in regions } i}{\#\text{cells in region } i}$$

where a *stable cell* means that all its neighbor cells are labeled. If all the cells for a certain region i are stable cells, it indicates that the node will not change anymore, hence $\xi_i = 1$.

Ideally the maturity of a region will tend to increase from 0 to 1.0 during the expansion of regions. A Markov process of *Beta* distribution with high concentration at the mode is used to demonstrate this relation, i.e.:

$$\begin{aligned} & P({}^t\xi_i \mid {}^{t-1}\xi_i) \sim \text{Beta}({}^t\xi_i \mid {}^{t-1}\xi_i; \alpha, \beta) \\ & \text{where } \beta = \begin{cases} \frac{\alpha - 1}{{}^{t-1}\xi_i} + 2 - \alpha, & \text{if } {}^{t-1}\xi_i \neq 0 \\ \alpha, & \text{otherwise} \end{cases} \end{aligned} \quad (6)$$

α is the parameter to define the sharpness of the *Beta* distribution. β is defined by taking the mode of the *Beta* distribution at $t^{-1}\xi_i$. A greater α indicates a faster evolution of the maturity α , namely faster to stop growing.³

V. LABEL UPDATE

According to section IV, the models of nodes can be updated separately. The label update process utilize the newly estimated model at iteration t as the real model for the node, i.e. ${}^tM_i \approx {}^t\hat{M}_i$. According to Bayesian theory, the probability distribution of the labeling over all the cells:

$$P({}^tL_n | {}^tM, {}^{t-1}L_n) \propto P({}^tL_n | {}^{t-1}L_n)P({}^t\hat{M} | {}^tL_n, {}^{t-1}L_n) \quad (7)$$

The second part is calculated by $P({}^t\hat{M} | {}^tL_n, {}^{t-1}L_n) = P({}^t\hat{M} | {}^tL_n)$, considering the Markov property. Notice that the update from current labeling tL_n to current model ${}^t\hat{M}$ is quite straightforward, if we consider each regional model represents the nature of each region that is defined by a distinctive label. As a result, the update of the models is represented as:

$$P({}^t\hat{M}_i | {}^tL_n) \sim N({}^t\hat{\mu}_i; {}^t\mu_i, \Sigma_i)N({}^t\hat{\xi}_i; {}^t\xi_i, \bar{\sigma}_i), \quad (8)$$

where Σ_i and $\bar{\sigma}_i$ are empirically small values.

Following algorithm 1, we assume the labeling of each single cell are only locally dependent. Considering an arbitrary cell in the map, we could see that it has maximum eight second ordered neighbors. The distributions of the labels of these neighbors are multinomial distributions, by which the label of the center cell is voted. In another word, the voting operation can also be seen as a sampling from the existing multinomial prior. The distribution is calculated from the number of cells that have the same label. It means that the sampling of the label for the center cell is subject to a Dirichlet prior summarizing the neighbor labels. This dependency is represented as follow.

$$P({}^tL_c | {}^{t-1}L_{cn}) \propto \prod_{m=1}^B \psi_m^{l_m-1} \quad (9)$$

where tL_c is the label of an arbitrary cell c at iteration t , ${}^{t-1}L_{cn}$ is the distribution of neighboring labels for cell c at $t-1$. B is the number of different labels among the neighbor cells, $B \leq K$. ψ_m is the normalized ratio of label m among the neighbor cells. l_m is a count of label m , as multinomial parameter.

By combining equation 9 8 and 7, the label of a cell is finally determined by maximizing a posterior (MAP) defined in equation 7.

VI. TESTS AND DISCUSSION

In application, the raw input are sparse indications from human input. We consider these indications as semi-supervisions. In another words, they define both the initial state of each cluster model. We remap these indications to the decomposed map and update ${}^{t=0}L_n$ accordingly.

³The results give in this paper use $\alpha = 5$.

A. Comparison

We compare the proposed algorithm with the result of seeded-Kmeans on the same dataset with same distance definition. A comparison result is shown in figure 4. Figure 4(a) depicts a targeting simulated indoor environment. The indications are direct manual inputs marked in blue color, which are sparse and arbitrary information. The seeded-EM result shown in figure 4(c) depicts how topological regions are distributed based on the Euclidean distances. The shortcoming is quite straightforward. The regions can not be correctly parted especially the Euclidean distance can not handle obstacles or blocked areas easily. The result using the proposed approach, in figure 4(e), has strong improvement in detecting compact regions while considering the maturity of regions.

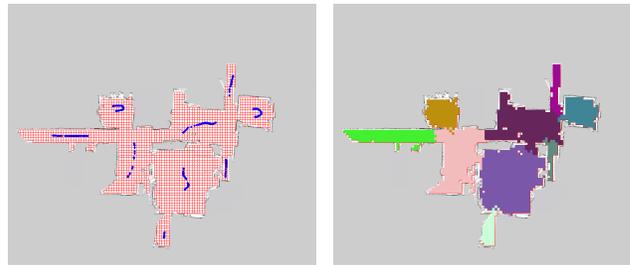


(a) Map and indication (b) seeded-Kmeans (c) our approach
Fig. 4. Result comparison

B. Results on real data-sets

From the given results, we could see that the proposed method could handle the compactness of the topological regions and human indications at the same time. As far as the search and rescue mission is considered, we would like to provide more results using real data-set in different environment. Figure 5 shows a segmentation result in a typical structured indoor environment. All the rooms and corridor are correctly segmented in the metric map based on the indication, shown in figure 5(c).

Figure 6 shows the result of a tunnel environment. The blue indications mark the space among crashed cars and wood tiles. We could see that the approach is able to create topological regions only based on sparse spatial indications.



(a) Raw map and indications (b) Result of this paper
Fig. 5. Result of a typical structured indoor environment.

VII. CONCLUSION

In this paper, we introduce a novel iterative semi-supervised clustering method using Markov process. We present its application in a topological segmentation problem

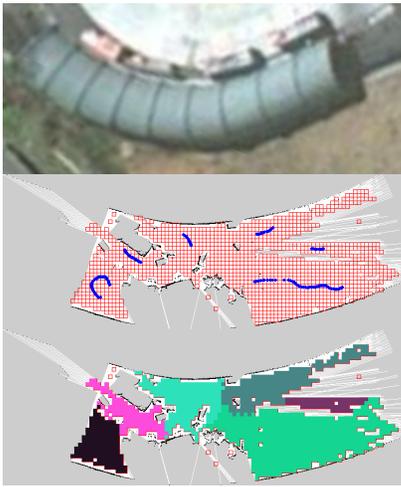


Fig. 6. Tunnel environment. From left to right: Google Map of the region; human indications; segmentation result.

based on sparse human indications. It is able to obtain a topological segmentation respecting sparse spatial indications, as well as adapting to the local structure and compactness from a metric map. Comparing to other methods, the proposed approach can better adapt to specific needs of a human user in different conditions.

It should be noted that this study has examined only the case of a known configuration space. This limitation is acceptable for off-line tasks such as mission level planning. As for incremental mapping, slam, and navigation tasks, the approach need to be adapted accordingly. For example, intermediate metric maps can also be used to generate local topological maps.

REFERENCES

- [1] C. Baber, RJ Houghton, R. McMaster, P. Salmon, NA Stanton, RJ Stewart, and G. Walker. Field studies in the emergency services. *HFI-DTC Technical Report/WP II. 1/1*, 1, 2004.
- [2] B. Balaguer, S. Balakirsky, S. Carpin, and A. Visser. Evaluating maps produced by urban search and rescue robots: lessons learned from robocup. *Autonomous Robots*, 27(4):449–464, 2009.
- [3] S. Basu, A. Banerjee, and R.J. Mooney. Semi-supervised clustering by seeding. In *Proceedings of the Nineteenth International Conference on Machine Learning*, pages 27–34. Morgan Kaufmann Publishers Inc., 2002.
- [4] S. Basu, A. Banerjee, and R.J. Mooney. Active semi-supervision for pairwise constrained clustering. In *Proceedings of the SIAM international conference on data mining*, pages 333–344, 2004.
- [5] M. Bilenko, S. Basu, and R.J. Mooney. Integrating constraints and metric learning in semi-supervised clustering. In *Proceedings of the twenty-first international conference on Machine learning*, page 11. ACM, 2004.
- [6] J. Blanc-Talon, D. Bone, W. Philips, D. Popescu, and P. Scheunders. *Advanced Concepts for Intelligent Vision Systems: 12th International Conference, ACIVS 2010, Sydney, Australia, December 13-16, 2010: Proceedings*. Springer, 2010.
- [7] J.L. Blanco, J.A. Fernández-Madrigal, and J. Gonzalez. Toward a unified bayesian approach to hybrid metric-topological slam. *Robotics, IEEE Transactions on*, 24(2):259–270, 2008.
- [8] J.L. Burke, R.R. Murphy, E. Rogers, V.J. Lumelsky, and J. Scholtz. Final report for the darpa/nsf interdisciplinary study on human-robot interaction. *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on*, 34(2):103–112, 2004.
- [9] Y. Chen, M. Rege, M. Dong, and J. Hua. Non-negative matrix factorization for semi-supervised data clustering. *Knowledge and Information Systems*, 17(3):355–379, 2008.
- [10] Jinwoo Choi, Minyong Choi, and Wan Kyun Chung. Incremental topological modeling using sonar gridmap in home environment. *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 3582–3587, October 2009.
- [11] H. Choset and K. Nagatani. Topological simultaneous localization and mapping (slam): toward exact localization without explicit localization. *Robotics and Automation, IEEE Transactions on*, 17(2):125–137, 2001.
- [12] A. Elfes. Using occupancy grids for mobile robot perception and navigation. *Computer*, 22(6):46–57, 1989.
- [13] U. Frese, P. Larsson, and T. Duckett. A multilevel relaxation algorithm for simultaneous localization and mapping. *Robotics, IEEE Transactions on*, 21(2):196–207, 2005.
- [14] B.J. Frey and D. Dueck. Clustering by passing messages between data points. *science*, 315(5814):972, 2007.
- [15] M. Gianni, P. Papadakis, F. Pirri, M. Liu, F. Pomerleau, F. Colas, K. Zimmermann, T. Svoboda, T. Petricek, GJM Kruijff, et al. A unified framework for planning and execution-monitoring of mobile robots. In *Workshops at the Twenty-Fifth AAAI Conference on Artificial Intelligence*, 2011.
- [16] N. Grira, M. Crucianu, and N. Boujemaa. Unsupervised and semi-supervised clustering: a brief survey. *A Review of Machine Learning Techniques for Processing Multimedia Content, Report of the MUSCLE European Network of Excellence (FP6)*, 2004.
- [17] Trevor. Hastie, Robert. Tibshirani, and JH (Jerome H.) Friedman. *The elements of statistical learning*. Springer, 2009.
- [18] B. Kulis, S. Basu, I. Dhillon, and R. Mooney. Semi-supervised graph clustering: a kernel approach. In *Proceedings of the 22nd international conference on Machine learning*, pages 457–464. ACM, 2005.
- [19] Benoit Larochelle, Geert-Jan M. Kruijff, and Jurriaan van Diggelen. Usage Patterns of Autonomy Features in USAR Human-Robot Teams. submitted to *FSR2012*, <http://www.nijfti.eu>.
- [20] T. Linder, V. Tretyakov, S. Blumenthal, P. Molitor, D. Holz, R. Murphy, S. Tadokoro, and H. Surmann. Rescue robots at the collapse of the municipal archive of cologne city: a field report. In *Safety Security and Rescue Robotics (SSRR), 2010 IEEE International Workshop on*, pages 1–6. IEEE, 2010.
- [21] Ming Liu, Francis Colas, and Roland Siegwart. Regional topological segmentation based on Mutual Information Graphs. In *Proc. of the IEEE International Conference on Robotics and Automation (ICRA 2011)*, Shanghai, 2011.
- [22] Z. Lu and T.K. Leen. Semi-supervised clustering with pairwise constraints: A discriminative approach. In *Proc. of the 11th International Conference on Artificial Intelligence and Statistics*, 2007.
- [23] T.P. McNamara. Mental representations of spatial relations* 1. *Cognitive Psychology*, 18(1):87–121, 1986.
- [24] L.Y. Morales Saiki, S. Satake, T. Kanda, and N. Hagita. Modeling environments from a route perspective. In *Proceedings of the 6th international conference on Human-robot interaction*, pages 441–448. ACM, 2011.
- [25] R.R. Murphy. Human-robot interaction in the wild: Land, marine, and aerial robots at fukushima and sendai. In *RO-MAN, 2011 IEEE*. IEEE, 2011.
- [26] B. Nadler and M. Galun. Fundamental limitations of spectral clustering. *Advances in Neural Information Processing Systems*, 19:1017, 2007.
- [27] J.B. Tenenbaum, V. Silva, and J.C. Langford. A global geometric framework for nonlinear dimensionality reduction. *Science*, 290(5500):2319, 2000.
- [28] S. Thrun. Learning metric-topological maps for indoor mobile robot navigation* 1. *Artificial Intelligence*, 99(1):21–71, 1998.
- [29] S. Thrun and A. Bücken. Integrating grid-based and topological maps for mobile robot navigation. In *PROCEEDINGS OF THE NATIONAL CONFERENCE ON ARTIFICIAL INTELLIGENCE*, pages 944–951. Citeseer, 1996.
- [30] S. Thurn, D. Fox, and W. Burgard. A probabilistic approach to concurrent mapping and localization for mobile robots. *Autonomous Robots*, 5:253–271, 1998.
- [31] E.A. Topp and H.I. Christensen. Detecting region transitions for human-augmented mapping. *IEEE Transactions on Robotics*, 26(4):715–720, 2010.
- [32] F. Wang, T. Li, and C. Zhang. Semi-supervised clustering via matrix factorization. In *Proceedings of The 8th SIAM Conference on Data Mining*, 2008.