Regional topological segmentation based on mutual information graphs

Ming Liu, Francis Colas, Roland Siegwart

Regional topological segmentation based on Mutual Information Graphs

Ming Liu, Francis Colas, Roland Siegwart
Autonomous Systems Lab, ETH Zurich, Switzerland
[ming.liu, francis.colas]@mavt.ethz.ch, rsiegwart@ethz.ch

Abstract—When people communicate with robots, the most intuitive mean is by naming the different regions in the environment. The capability that robots are able to identify different regions highly depends on the unsupervised topological segmentation results. This paper addresses the problem of segmenting a metric map into regions. Nowadays many researches in this direction develop approaches based on spectral clustering. However there are inherent drawbacks of spectral clustering algorithms. In this paper, we first discuss these drawbacks using several testing results; then we propose our approach based on information theory which uses Chow-Liu tree to segment the composed graph according to the weight differences. The results show that our method provides more flexible and faster results in the sense of facilitating semantic mapping or further applications.

I. INTRODUCTION

MAPPING is a method of environment modelling, whose outcome is a representation of the surroundings, namely a map. Applicable maps are usually categorized into metric maps, topological maps, hybrid maps, semantic maps etc. Specifically, topological maps consist of nodes with edges between the nodes. The topological mapping techniques can be classified more precisely according to what the nodes and edges represent. For visual navigation, usually the "nodes" represent the waypoints the mobile robot should follow, and the "edges" are the physical transitions between nodes. On the other hand, for the reasoning of an environment or achieving a semantic representation of the environment, the "nodes" can represent functional regions, such as rooms. In this paper we will discuss the methods of creating the topological maps on which appear the relations between different regions. By "topological map", we will designate a representation of the relations between regions, such that a "node" on the map represents a region of traversable free space, and an "edge" represents the linkage between two of them.

Compared to a metric map, a topological map is suitable for a better quality of human-robot interactions. To a certain extend, topological mapping is the way used by human brains in everyday life. Moreover, in comparison with the metric maps, topological maps are more compact in the sense of storage saving and scalability. For these reasons, several works related to semantic mapping [1] [2] or cognitive mapping [3] used topological map as the basic representation of the environment. In this paper we will show that our method is able to produce a relatively stable topological segmentation of

the map, which can be potentially used for further environment modelling.

Several representation methods or descriptors use vision for the topological nodes, such as fingerprint of places [4] and adaptive color tags of places [5]. These descriptors can work under the condition that vision approach provide nourishing raw information from the environment. The use of laser scan cannot provide as much direct information as vision. Therefore several methods treat a bunch of aligned consequent 2D scans as a subimage, then perform the image-based segmentation based on vision techniques, such as [1] uses watershed to segment the given metric map. There are other approaches that directly use a part of the occupancy grid map as a submap. This manner is considered as the most basic and efficient representation of a node, and got widely applied, such as in [6] [7]. As for how maps are constructed, several representation methods can be introduced [8], e.g. Voronoi diagram [9].

In this work we use structured cells to represent the basic decomposition of the global map as a connected graph. Generally, the basis of topological mapping is topological segmentation, which itself is based on the space decomposition. The decomposition result will directly affect the outcome of topological mapping, since it defines the basic units that will combine together to construct a topological region. Several methods of space decomposition are used in the computer vision community, such as Quadtrees, K-D trees, Octrees etc. Considering the repeatability and extendibility, our approach is based on the decomposition resulting from a Quadtree structure.

Among the state-of-art works, spectral clustering [10] has been widely used in topological mapping. [11] takes key positions during the metric mapping process, builds the adjacent matrix based on the distances between these key positions, then use spectral clustering to segment the graph in an online manner. The authors of [7] used spectral clustering as an offline method on the global map. This work was followed by an incremental mapping and classification method presented in [12]. Despite the wide utilization, the spectral clustering method requires the calculation of inverting a big matrix (the size depends on the number of nodes in the graph), which makes it quite expensive in computation. Besides, Nadler et al in [13] stated that the spectral clustering has its 'fundamental' limitations, e.g. the scalable problem. In this paper, we will provide an exemplar map following the idea from Nadler, to show that the spectral clustering would not

have a rational result out. Nevertheless, as a side-contribution to the community, we will also present an approach based on spectral clustering in Sec. II, using a simple distance function to construct the adjacent matrix.

Considering the drawbacks of the spectral clustering method, we will also propose a new topological segmentation method based on *Shannon information theory*, using mutual information as the indicator of the relation between two adjacent nodes. This work is stimulated by the use of mutual information in the computer vision community, such as [14]. To our knowledge, there are no direct references related to topological mapping using information theory. The most relevant work is the active search scenario introduced by Davison [15]. Margarita et al extended his work to the inferring of the hierarchical structure of visual maps in [16]. In this paper, we will define the mutual information used in the decomposition space. To segment the graph-like mutual information tree, a common approach is to use Chow-Liu tree to build the spanning tree, since Chow and Liu showed in their 1968 paper [17] that a full, joint probability density function can be optimally approximated as a product of second-order conditional and marginal distributions, which will minimise the difference in Kullback-Leibler divergence [18]. The segmentation of the tree can then be done by the selection of the level to which Chow-Liu tree expands.

The techniques such as the efficient segmentation, introduced by Mezaris et al in [19], can also be used in the segmentation of a map. According to our test, it has issues with repeatability, and is sensitive to parameter changes. Regarding the topological segmentation problem, it may not be an effective solution. The result will not be included here.

As an application of our approach, the following procedure can be recommended. An global map based on occupancy grid, which can be obtained by laser or sonar, is firstly required, . Then the free space in the global map will be decomposed into small cells using Quadtree structures. These small cells are related to one another by mutual information trees. Finally, the decomposition of the information trees will determine different topological regions.

We would like to stress the following possible contributions of this paper:
- It provides a simple method creating the adjacent matrix for using spectral clustering in topological segmentation.
- Inspired by Nadler's work [13], we point out the inherent limitations of spectral clustering in map segmentation.
- Despite that several existing works in the computer vision community are using mutual information to do image segmentation and matching, we are the first to propose a topological mapping approach based on information theory.

As a detailed explanation of the introduction, the remainder of this paper is organized as follows. The spectral clustering approach and its limitations in topological mapping will be discussed in the next section; then in Sec. III, we will explain our representation of the map based on information theory in detail; further on we will present different methods of factorization and segmentation of the tree. Test results and conclusion will be given at the end.

II. SEGMENTATION BASED ON SPECTRAL CLUSTERING AND ITS LIMITATION

Usually, the raw maps that we achieve through SLAM algorithm are in the form of occupancy grids. There are commonly three states of occupancy: unknown, occupied and free space. Getting a topological segmentation of the map amounts to the segmentation of the free space. In order to simplify the problem, we simulated a typical indoor map shown in Fig. 1(a). With the purpose of getting a more general representation, a space decomposition method is firstly performed on the raw map. In this work we choose Quadtree as the basic strategy of space decomposition. It requires an input parameter to define to which maximum depth we build the tree. The major depth means the most precise segmentation of the map. Each cell in the decomposition result is called a 'node' in the remainder of this paper.

The general algorithm of spectral clustering requires the neighbourhood graph together with the corresponding *adjacency matrix W* with $n \times n$ elements w_{ij}, where n is the number of nodes in the graph. For various methods in this area, a big difference is the definition of the weight w_{ij} between nodes. In our work, we define the weight between two nodes as follows:

$$w_{ij} = \begin{cases} l_{co}(i,j), & \text{if i and j are adjacent} \\ 0, & \text{otherwise} \end{cases} \quad (1)$$

where $l_{co}(i,j)$ is the length of the common edge between the nodes i and j. We only consider the 1st order neighbourhoods, i.e. the diagonal neighbours are not included. Following the notation of Von Luxburg in [10], the algorithm of spectral clustering is shown in Alg. 1

Algorithm 1: Spectral Clustering on Topological Segmentation

Input:
Adjacency matrix W, where w_{ij} indicates the weight between two nodes i and j;
The request number of clusters k
$T = \{T_1, T_2, \dots T_k\} \simeq \{1, 2, \dots, k\}$
Output:
The list of indices corresponding to the nodes,
1 Calculate the normalized graph Laplacians using $L_{sym} := I - D^{-1/2}WD^{-1/2}$ or $L_{rw} := I - D^{-1}W$, where $D = \text{diag} d_1, \dots, d_n$ and $d_i = \sum_{j=1}^{n} \omega_{ij}$;
2 Calculate the k smallest eigenvectors $u_1, \dots, u_k$ of L (either L_{sym} or L_{rw}), form the matrix $U = [u_1 \dots u_k] \in \mathbb{R}^{n \times k}$
3 Set $\hat{U}$ to be U with rows re-normalized to have unit norm, by $\hat{U}_{ij} = \frac{U_{ij}}{\left(\sum_j U_{ij}\right)^{\frac{1}{2}}}$
4 Use k-means clustering on the rows of $\hat{U}$, into k clusters
5 Assign label T_i to cluster j if and only if row j of $\hat{U}$ is assigned to cluster j.

A. Sensitivity to k

The decomposition with different levels and results for the topological segmentation on the simulated environment is shown in Fig. 1(b) and Fig. 1(c). We could see from the results that two factors will affect the results much. The first one is the decomposition level, i.e. sometimes the shallower level will naturally and directly divide the separated regions, such as the dark green cluster number 6 in Fig. 1(b). The second factor is the number of clusters. We intentionally choose 8 as the number of clusters for the test in Fig. 1(c). A redundant cluster (marked with dark blue) is detected without obvious reasons.

(a) Raw map   (b) Result at level 6   (c) Result at level 7
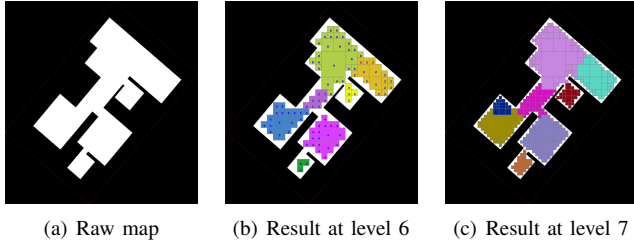
Fig. 1.   Segmentation results based on spectral clustering

This can be considered as the first limitation of the spectral clustering method, since for most applications we have neither information on the number of regions in a global map, nor rough ideas of the optimal size of the cells.

B. Unrepeatable results

Another result of spectral clustering is achieved by processing the map provided by Choi et al in [7]. The result is shown in Fig. 2. Our own result appears in Fig. 2(b). Both results in Fig. 2(a) and 2(b) are acceptable in the general sense of "separating rooms". However, with the same parameters, sometimes we can also obtain other results such in Fig. 2(c): this segmentation somehow makes sense from the raw sensory data aspect, but is less useful for environment modelling e.g. semantic mapping and reasoning. These results imply the instability of similar approaches. The reason can be partially inferred from the essence of the spectral clustering.
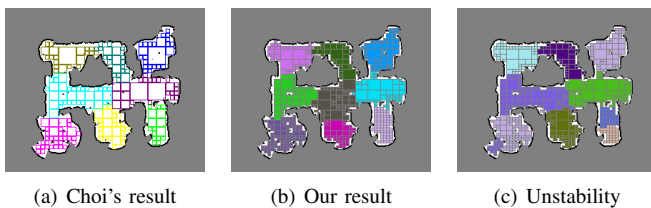

(a) Choi's result   (b) Our result   (c) Unstability

Fig. 2.   Comparison of spectral clustering with different distance functions

C. An extreme case

Moreover, under extreme cases, the spectral clustering cannot even rationally work. Following the point cloud example provided in [13], we create a simulated environment

shown in Fig. 3(a). The environment is constructed by a rectangle 'corridor' area and circler 'room'. The previous spectral clustering will achieve a result shown in 3(b), and other spectral clustering methods with different weighting functions provide similar results: the 'room' and the 'corridor' cannot be correctly segmented. Then we increased the k, hoping that the segmentation can randomly hit the 'room'. The result of $k=3$ is shown in Fig. 3(c). It still can not find the rational segmentation. This phenomenon was called the "scaling problem" of the spectral clustering in [13]. It means that we cannot simply increase the k to approach the result that we are looking for. In fact, there are no feasible ways to control the segmentation result using spectral clustering.

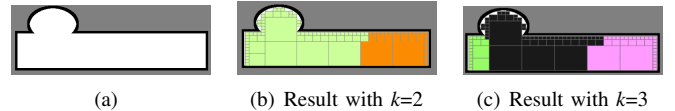(a)   (b) Result with $k=2$ (c) Result with $k=3$

Fig. 3.   Example map showing the limitation of spectral clustering

Beside the listed drawbacks, there are other issues such as the fact that it is hard to combine several submaps in order to form a compact union while the tree structure of each segmented region is not clear. All in all, a more sophisticated method is required to tackle the topological segmentation problem. We will introduce our advanced approach in the following section.

III. Mutual Information Tree and factorization

Regarding the segmentation of the whole graph described in Fig. 1(a), the essence of the segmentation problem is to cluster the cells into several groups. When we investigate the result of the segmentation, we could also consider that one typical cluster is formed by a few center cells surrounded by its "supporters", who vote for a new cluster together with the center cells. According to the decomposition result shown in Fig. 1(a), each cell has its direct neighbours, which can be taken as the direct "supporters". However, because the weights of "supporters" are different, some weak linkages are cut off. It leads to a natural segmentation between clusters.

Considering this mechanism, we introduce Shannon information theory to represent the relations inside each cluster. The 'supporting strength' of the "supporters" is the information entropy according to the length of the common edges between the center cell and its direct neighbour. Using the notation of Mackay [20], and Davison [15], the direct neighbours of a center cell X lie within the discrete 'alphabet' $A_X = \{a_1, a_2, \dots\}$. It is represented by mutually exclusive statements of the form '$x = a_i$', assigned probabilities $P(x = a_i)$ which sum to one. The probability $P(x)$ indicates the exclusive "supporting" ratio from a certain direct neighbour of the center cell. It means that the bigger weight of the common edges implies a "stronger" support.

The *information entropy* $H(X)$ of this distribution is:

$$H(X) = E\left[\log_2 \frac{1}{P(x)}\right] = \sum_{x \in A_x} P(x) \log_2 \frac{1}{P(x)}, \qquad (2)$$

$P(x)$ is the probability that the support is provided by x, such as

$$P(x) = \frac{l_{co.x}}{\sum_{x \in A_x} l_{co.x}} \qquad (3)$$

where $l_{co.x}$ is the length of the common edge between a_i and X. Here we use shorthand $P(x)$ for $P(x = a_i)$. $H(x)$ is then a measure of the average supporting strength around a center cell X in bit units.

For two neighbour cells, the conditional probability should also be considered. $P(longerlengthX|Y)$ will represent the possibility of "X getting support among the existing "supporters" of Y. Therefore, the entropy:

$$H(X|Y) = E\left[\log_2 \frac{1}{P(x|y)}\right] = \sum_{x \in A_x} P(xy) \log_2 \frac{1}{P(x|y)}, \qquad (4)$$

$P(xy)$ is related to the common neighbours of X and Y.

This leads to the *mutual information $I(X;Y)$* as follows:

$$I(X;Y) = H(x) - H(X|Y) \qquad (5)$$

Following the expression in [15], it indicates the average expected reduction in entropy of one cell, by learning the supporting status of one of its neighbours. Therefore, $I(x,y)$ indicates how much exclusive support one cell X can get, considering its neighbour Y. When this value is big, it means the connections between X and Y are close, in the sense that they form a compact cluster.

We can therefore define a *mutual information* matrix as in Eq. 6, where N is the total number of nodes on the map.

$$I = \begin{pmatrix} * & I(x_1;x_2) & \ldots & I(x_1;x_N) \\ I(x_2;x_1) & * & \ldots & I(x_2;x_N) \\ \vdots & \vdots & \vdots & \vdots \\ I(x_N;x_1) & I(x_N;x_2) & \ldots & * \end{pmatrix} \qquad (6)$$

The matrix is symmetric and the diagonal elements are meaningless while they represent the connection weight to itself.

The plot of *mutual information* for the simulated environment Fig. 1(a) is shown in Fig. 4(a).
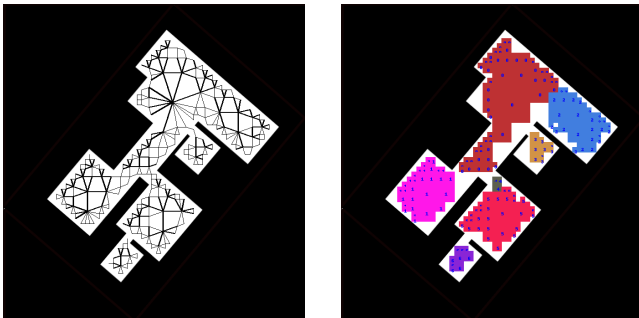
(a) The plot of *mutual information*. The thickness of edges represent the magnitude of *mutual information*.

(b) Segmentation result at $th = 2.0$.

Fig. 4. The representation of the *mutual information* graph and the segmentation result.

IV. TREE FACTORIZATION AND CHOW-LIU TREE

The *mutual information* graph we got in the previous section is a nonspanning graph. There are cycles in the graph structure. To generate a spanning tree from the graph, the usual way is to use Chow-Liu tree factorization. The essence of Chow-Liu tree is to be a *Maximum Weights Spanning Tree*. We build the Chow-Liu tree by following the standard procedure, using the *mutual information* as the weights of edges. The tree-building starts from the maximum weighted edges. At each step, the edge with the biggest weight will be added to the tree, if and only if the edge does not form any cycles with the existing edges. This process will stop when all the nodes are connected, returning a fully connected Chow-Liu tree.

As for the application of Chow-Liu tree on a real dataset, the *mutual information* graph is not likely to be a full connected tree. We need to separate the unconnected sub-graphs first. The algorithm can be stated as follows (Alg. 2):

Algorithm 2: Search all sub-graphs in the graph

Input:
The set of all the nodes in the graph: $Ni = 0$
Output:
Separated sets of connected nodes: N_i

1 **while** $N! = \phi$ **do**
2   Define a random node $n_i \in N$ as a seed node
3   Find all the nodes connected to the seed node using Breadth First Search, and save them as set N_i
4      $N = N \backslash N_i$, remove the N_i subset from N
5      $i = i + 1$

For each N_i, at least one region should be kept. In other cases, the factorization process will stop by a certain threshold level, i.e. the weak links whose weights are under the predefined threshold level will be ignored.

We build the spanning tree over the extreme case shown in 3, on which the spectral clustering fails. The segmented result is shown in Fig. 4(b). As a natural *mutual information* graph, we could see that the free space in the map is segmented into 3 parts. If we ignore the weak weighted edges under 2.0, the result is shown in Fig. 4(b). Although it cannot give exactly the expected segmentation, the 3-node structure is reasonable in the sense of regional detection.
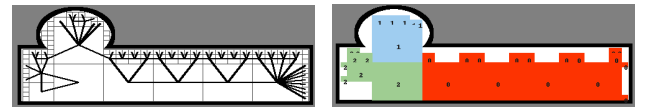
(a) *mutual information* graph   (b) Segmentation result at $th = 2.0$

Fig. 5. Segmentation result on the extreme case

V. RESULTS ON METRIC MAPS

A. Functional Test

In order to test our method on a larger scale with real data input, we carried out a test on the occupancy grid map

created in an office environment. The map covers an area of 50×18 square meters. The structure of the environment is quite complex, including doorways, corridor and office rooms etc. The segmentation results are shown in Fig. 6.

With a low threshold level such as 1.0, the proposed method can get roughly the same result as the spectral clustering based method (the one shown in Fig. 6(a)). Sometimes the occupancy map is too big in the sense of semantic mapping, i.e. one whole office room is too hasty for the reasoning of the environment. Take the region number 6 (light brown) in Fig. 6 as an example: there are separated parts inside the room. How could we do a further segmentation on that? As for spectral clustering, Fig. 3(b) and (c) have already given the answers: the spectral clustering doesn't have the ability to provide the rational segmentation the environment at a more precise level based on the number k. This is again due to the 'scale' problem which is caused by its nature. On the contrary, the proposed method based on the *mutual information* graph can achieve the result shown in Fig. 6(b).

As a matter of fact, there are small regions which can be considered as segmentation noise in both cases. These noisy regions can be eliminated by median filters. This assumption can be inferred from the plot of the *mutual information* matrix I, as shown in Fig. 7. To simplify the expressions, we use the simulated map in Fig. 3 as an example. We can see that
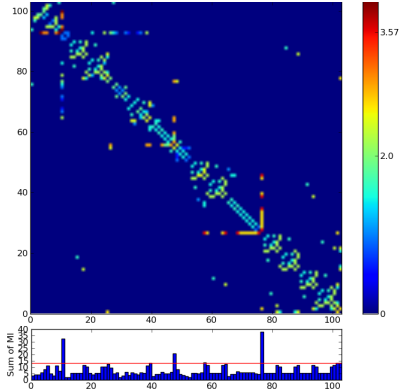

Fig. 7. The plot of the *mutual information* matrix *I* of the map in Fig. 3. The greater value means stronger connections between two nodes.

most of the high value sites are lying on the diagonal of the matrix *I*. It indicates that most of the 'support' comes from the neighbours, which fits our assumption in the definition of the *mutual information*. This plot also implies the potential "centers" of each cluster. If we accumulated the weights values that lie in the same row or column, the result represents the impact of a certain node when a cluster is generated around it. For example, in the Fig. 7, we can intuitively see three dominating "centers" in the histogram.

Combining the tree structure and the *mutual information* matrix, the result from the *mutual information* graph can be easily controlled by selecting different threshold levels, since the higher threshold level will provide more regions and the strong links will come together with high accumulated

weights. On the contrary, the result of spectral clustering is not intuitively controllable; once the user requires more sub-regions, the k-means clustering needs to be run since the beginning, which requires a sequence of test-and-run processes.

B. Computational Complexity

The time consumption of the primary steps is shown in Fig. 8. This test uses both methods -*mutual information* graph and
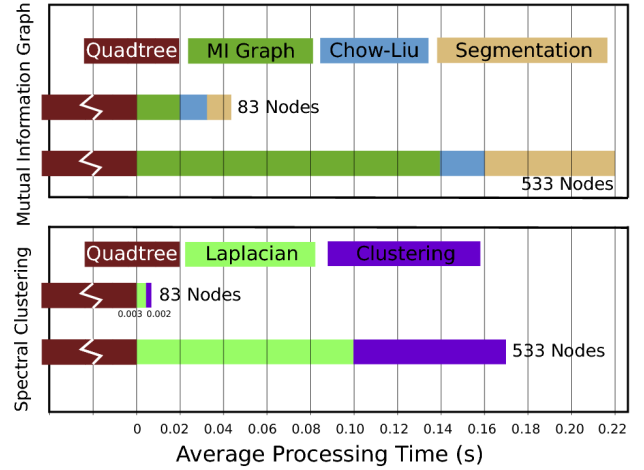

Fig. 8. A comparison of the computational complexity

spectral clustering- to segment two different maps. Because the difference of scale, the two maps have different number of nodes. In the procedure of both methods, the Quadtree works as the basic space decomposition method. This part of the process can be replaced by another decomposition method and the time consumption is proportional to the maximum depth of the tree with the computational complexity of $O\left(n+1\right)^2$, where *n* is the maximum depth of the tree[1]. This decomposition process is not closely related to the clustering methods, so that the calculation of the time consumption is started after deomposition.

As for the optimization of the computational complexity, we did some pre-processing to obtain the adjacent matrix. We calculate all the neighbour relationships separately and save them in an indexed table. As a result, for the *mutual information* calculation in Eq. 5, the process only takes $O(n)$ complexity instead of $O(n^2)$ complexity (for n nodes searching neighbours among n nodes). The result in Fig. 8 confirmed this assumption: the second map has around 7 times more nodes than the first map (533:83). We can see that the "MI Graph" (short for *mutual information* graph) process consumes roughly 7 times more time for the second map.

Compared to the proposed method, the spectral clustering one is much faster when the graph is small. However, as the matrix inversion is the major step in the calculation of the

---

[1]A typical processing time for Quadtree decomposition on a laptop with 2.5GHz CPU is around 1.4 seconds for the depth of 7.
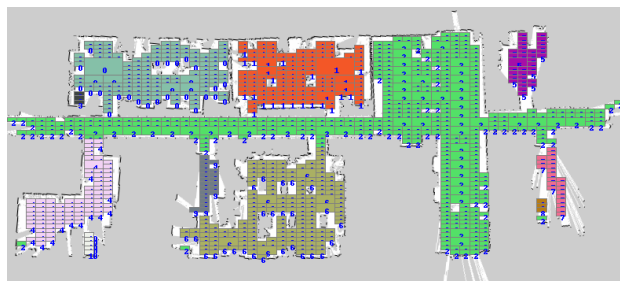
(a) Segmentation by optimized spectral clustering
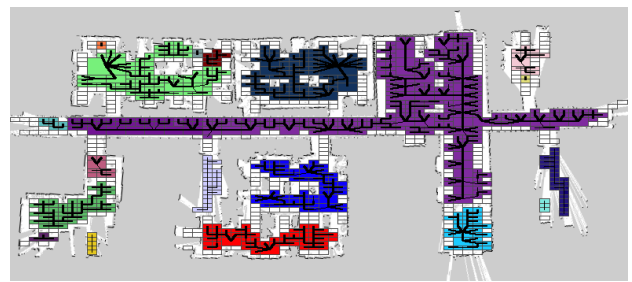
(b) Segmentation by mutual information at $th = 1.6$. The noisy small regions can be eliminated by median filters

Fig. 6.   Segmentation of a metric map

Laplacians, which holds $O(n^3)$ complexity, the processing time dramatically rises when it comes to the second map.

According to the result in Fig. 8, our method shows an important timing features. The 'lightweight' characteristic is helpful and efficient in the real-time mobile robot mapping and navigation applications.

VI. CONCLUSION

In this paper, we first gave a new segmentation method based on the widely cited spectral clustering theory. By comparing our results with others, we showed the inevitable drawbacks of topological segmentation methods which are based on spectral clustering. We then proposed a segmentation method based on *information theory*. This method is derived from the *mutual information* graph and Chow-Liu tree factorization. We stated and certified the following characteristics of the proposed method:

1) It is fully scalable. The segmentation result can be easily optimized by thresholding weights on local sub-graphs.
2) The segmentation results are Stable. The Chow-Liu tree factorization is unique under the same weight function, which avoids the randomness introduced by the *k*-means clustering method used in spectral clustering.
3) It is computational cheap in large scale: we have showed that the method can work under $O(n)$ complexity.

The proposed method would be well employed in a number of real use cases, such as a rescue mission for example. After a fire, the blueprint of the building is not directly accessible or needs time to get. The segmentation based on the metric map would provide intuitive information to the rescuers, who could use it as a basis to define determined regions as targets to be assigned to the firemen.

REFERENCES

[1] C. Galindo, A. Saffiotti, S. Coradeschi, P. Buschka, J. Fernandez-Madrigal, and J. Gonzalez, "Multi-hierarchical semantic maps for mobile robotics," aug. 2005, pp. 2278 – 2283.
[2] O. M. Mozos, A. Rottmann, R. Triebel, P. Jensfelt, and W. Burgard, "Supervised semantic labeling of places using information extracted from sensor data," *Robotics and Autonomous Systems*, vol. 55, pp. 391–402, 2007.
[3] "Cognitive maps for mobile robots–an object based approach," *Robotics and Autonomous Systems*, vol. 55, no. 5, pp. 359 – 371, 2007, from Sensors to Human Spatial Concepts.
[4] A. Tapus and R. Siegwart, "Incremental robot mapping with fingerprints of places," in *Intelligent Robots and Systems, 2005.(IROS 2005). 2005 IEEE/RSJ International Conference on*, 2005, pp. 2429–2434.
[5] M. Liu, R. S. Davide Scaramuzza, Cédric Pradalier, and Q. Chen, "Scene recognition with omnidirectional vision for topological map using lightweight adaptive descriptors," in *IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS 2009*, 2009.
[6] S. Simhon and G. Dudek, "A global topological map formed by local metric maps," in *IEEE/RSJ International Conference on Intelligent Robotic Systems*, vol. 3.   Citeseer, 1998, pp. 1708–1714.
[7] J. Choi, M. Choi, and W. K. Chung, "Incremental topological modeling using sonar gridmap in home environment," pp. 3582–3587, Oct.
[8] "Learning metric-topological maps for indoor mobile robot navigation," *Artificial Intelligence*, vol. 99, no. 1, pp. 21 – 71, 1998.
[9] H. Choset, "Sensor based motion planning: The hierarchical generalized Voronoi graph," Ph.D. dissertation, Citeseer, 1996.
[10] U. V. Luxburg, "A tutorial on spectral clustering," *Statistics and Computing*, no. August 2006, pp. 395–416, 2007.
[11] E. Brunskill, T. Kollar, and N. Roy, "Topological Mapping Using Spectral Clustering and Classification," in *International Conference on Intelligent Robots and Systems*.   IEEE, 2007, pp. 3491–3496.
[12] J. Choi, M. Choi, and W. K. Chung, "Incremental topological modeling using sonar gridmap in home environment," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*.   Ieee, Oct. 2009, pp. 3582–3587. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=5354247
[13] B. Nadler and M. Galun, "Fundamental limitations of spectral clustering," *Advances in Neural Information Processing Systems*, vol. 19, p. 1017, 2007.
[14] J. Kim, J. Fisher III, A. Yezzi, M. Çetin, and A. Willsky, "A nonparametric statistical method for image segmentation using information theory and curve evolution," *IEEE Transactions on Image Processing*, vol. 14, no. 10, 2005.
[15] A. J. Davison, "Active Search for Real-Time Vision," in *ICCV*, 2005, pp. 66–73 Vol.1.
[16] M. Chli and A. J. Davison, "Automatically and efficiently inferring the hierarchical structure of visual maps," *2009 IEEE International Conference on Robotics and Automation*, pp. 387–394, May.
[17] C. Chow and C. Liu, "Approximating discrete probability distributions with dependence trees," *IEEE transactions on Information Theory*, vol. 14, no. 3, pp. 462–467, 1968.
[18] S. Kullback and R. Leibler, "On information and sufficiency," *The Annals of Mathematical Statistics*, vol. 22, no. 1, pp. 79–86, 1951.
[19] V. Mezaris, I. Kompatsiaris, and M. Strintzis, "A framework for the efficient segmentation of large-format color images," in *Proc. International Conference on Image Processing*, vol. 1.   Citeseer, 2002, pp. 761–764.
[20] D. MacKay, *Information theory, inference, and learning algorithms*. Cambridge Univ Pr, 2003.