HAL
open science

Analysis of methods for reducing line segments in maps: Towards a general approach

Francesco Amigoni, Simone Gasparini

To cite this version:

Francesco Amigoni, Simone Gasparini. Analysis of methods for reducing line segments in maps: Towards a general approach. IROS 2008 - IEEE/RSJ International Conference on Intelligent Robots and Systems, Sep 2008, Nice, France. pp.2896-2901, 10.1109/IROS.2008.4650847 . inria-00435547

HAL Id: inria-00435547

https://inria.hal.science/inria-00435547

Submitted on 26 Nov 2009

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Analysis of Methods for Reducing Line Segments in Maps: Towards a General Approach

Francesco Amigoni and Simone Gasparini

***Abstract*— Segment-based maps are emerging as an efficient way to represent the environments in which mobile robots operate. When compared to grid-based maps, maps composed of line segments usually need less space to be stored. However, very little effort has been devoted to methods that allow to reduce the size of segment-based maps by removing redundant line segments that represent the same object in the environment. This problem is usually addressed with rather *ad hoc* methods that are embedded in mapping systems. In this paper, we put forward the problem of reducing the size of segment-based maps by presenting a survey of the existing methods and by experimentally evaluating some of them. Our results can be used to set out some guidelines for the development of a general approach to reducing redundant line segments in maps.**

I. INTRODUCTION

Techniques for robot map building have obtained a considerable attention in the last years [1]. More recently, a number of researchers advocated the use of line segments as a way to reduce the size of the data structures storing maps with respect to the widely used grid-based maps [2], [3], [4], [5]. In order to use them in efficient path planning and localization processes, it is important to limit the size of segment-based maps by eliminating the redundant line segments representing the same portion of the environment. Redundant line segments usually appear because a robot acquires several times data about the same part of the environment (e.g., for localization purposes) and because, in multirobot settings, more robots acquire data about the same part of the environment. Thus, the problem that arises is that of "fusing" together the line segments that represent the same object in the environment. So far, this problem has not been convincingly addressed by the robotic mapping community. Only *ad hoc* methods that depend on particular sensors [3] or that work with highly-structured environments [6] have been embedded in mapping systems.

This paper aims at providing some contributions towards the definition of a general method for reducing the number of redundant line segments in maps. In particular, we survey the most significant methods that have been proposed to simplify segment-based maps and we experimentally compare some of these methods. We selected a significant sample of methods, we applied them to different data sets, and we evaluated the results both quantitatively and qualitatively. From this analysis some guidelines emerged to drive the definition of a general method for fusing line segments in maps. Hence, the original contribution of this paper is not the proposal of new methods for fusing line segments in maps but the attempt to critically survey and compare the existing methods, towards the definition of more general approaches.

This paper is structured as follows. The next section introduces the problem of reducing the number of redundant line segments in the general framework of robotic mapping. Section III surveys the methods that have been proposed in literature for simplifying segment-based maps. In Section IV we experimentally compare some of these methods. Finally, Section V concludes the paper by discussing the possible impact of the obtained results on the definition of a general method for reducing the number of line segments in maps.

II. PROBLEM DEFINITION

Generally speaking, a system for building two-dimensional segment-based maps of environments using a mobile robot operates in the following steps.

(a) Acquisition of a new *scan* S^t (e.g., a set of range measurements) from the current position p^t of the robot in the environment (t is the current time step) and approximation of the data contained in the scan with line segments (see [7] for a survey of methods that approximate points acquired withy laser range scanner with lines).

(b) Registration of S^t, namely alignment of the newly acquired scan with the *current map* M^t. In many cases, the registration is performed between S^t and S^{t-1}. However, we consider M^t because it contains information about the older scans and, in particular, about S^{t-1}. Registration usually amounts to finding a rototranslation r^t (i.e., a translation and a rotation in the plane) that is applied to the line segments in S^t to align them to the line segments in M^t. Some approaches use data from odometry to have an initial estimate of r^t [8], some other approaches use only geometrical information in S^t and M^t to find r^t [9].

(c) Addition of the line segments of S^t (after registration) to M^t to obtain the *updated map* M^{t+1}. In order to keep in memory the minimum necessary number of line segments, redundant line segments are fused together.

(d) Determination of the next observation position p^{t+1} for the robot and travel to p^{t+1}. Then, the process starts again from (a).

The above mapping process can be extended, with minor modifications, to other situations, including the case in which the line segment approximation of a scan is performed after

F. Amigoni is with the Dipartimento di Elettronica e Informazione, Politecnico di Milano, Milano, Italy. francesco.amigoni@polimi.it.

S. Gasparini is with INRIA Rhône-Alpes, Grenoble, France. simone.gasparini@inrialpes.fr.

registration and the case in which multiple robots collect scans and send them to a centralized server.

In many systems presented in literature, step (c) just adds the line segments in S^t to those in M^t, without attempting any fusion. In systems in which fusion is performed, step (c) is usually addressed by rather *ad hoc* solutions, as we detail in the next section. To get some insights and to compare them, we propose the following general framework for describing the methods that fuse line segments in step (c).

(α) Find a set L of line segments that represent the same object o in the environment;

(β) Find a new set $\hat{L}$ of line segments that substitute L in representing the object o; of course $|\hat{L}| \leq |L|$, namely $\hat{L}$ contains less line segments than L. Then, the process starts again from (α) until no new L can be found.

Step (α) is usually based on a set of conditions that line segments must satisfy in order to be inserted in L. We call them *fusion conditions*. Note that the line segments in L could come either from the newly acquired scan S^t or from the current map M^t. Note also that fusion conditions are related to the problem of data association, that we do not further discuss here because it is mainly outside the scope of this paper. Step (β) is usually based on a *fusion algorithm* that prescribes how to obtain $\hat{L}$ from L. Note that the line segments in $\hat{L}$ are inserted in the updated map M^{t+1}. The methods for simplifying segment-based maps differ in the fusion conditions and the fusion algorithms they employ. According to this framework, in the next section, we survey some of these methods, concentrating on those that operate at the level of line segments, not considering the raw data and the sensors with which they have been acquired.

III. A SURVEY OF EXISTING METHODS

To the best of our knowledge there is no specific work exclusively devoted to reducing the number of line segments in a map (except [10]). As said, more often than not, mapping systems simply do ignore the problem. When they do not, *ad hoc* procedures are used. Although usually these procedures strongly depend on the characteristics of the corresponding mapping system (e.g., on the particular sensors, registration procedures, or line segment representation adopted), in the following survey we try to abstract from details to expose the basic ideas on which the fusion procedures are based. Our discussion roughly goes from basic to complex fusion procedures.

A. Basic Fusion Procedures

Some approaches apply very basic (and limited) fusion procedures. For example, in [11] line segments that are collinear according to the sensor's resolution (fusion conditions (α)) are fused into a single line segment (fusion algorithm (β)). In other cases ([5], [12]), some line segments are merged, but full details are not provided.

B. Fusion Procedures for Pairs of Line Segments

A number of fusion procedures fuse a *pair* of line segment in a new *single* line segment. Namely, for these methods, $|L| = 2$ and $|\hat{L}| = 1$.

For example, in [13], the fusion conditions are defined according to the concept of "viewing sector": given a line segment, a viewing sector is defined as the region within the scanning rays (from sensor position) of its endpoints. A line segment s is considered to correspond to another line segment s' (one from M^t and one from S^t) when s' lays in the viewing sector of s and the endpoints of s' are close enough to s. When two line segments s and s' correspond, they are merged in a new line segment with a fusion algorithm that combines the parameters of s and s'. In [6], two line segments are selected for fusion and are substituted by a third line segment. The details of the fusion conditions and of the fusion algorithm depend on the fuzzy segment representation used by the authors. The main drawback of their approach is that it is suitable only for highly structured environments with perpendicular walls. Another approach [14] uses a statistical representation of the line segments that takes into account the fitting errors of the raw range data (points). According to this representation, two line segments match if their statistics satisfy the chi-squared test. Matching line segments are replaced by a third line segment using a maximum likelihood approach. Also the method proposed in [10] fuses pairs of line segments that (supposedly) represent the same portion of the environment. We detail this method in Section IV-A.

C. Fusion Procedures for Sets of Line Segments

Some fusion procedures try to fuse a *set* of line segments in a new *set* of line segments. For these methods, $|L| \geq 2$ and $|\hat{L}| \geq 1$.

In [9], close line segments are grouped in matching chains that are approximated by polylines. This method will be illustrated in more detail in Section IV-A. A similar approach is presented in [15]: also in this case multiple line segments are fused together. The fusion conditions build clusters of line segments that are parallel, on the same hyperplane, and close to each other. The fusion algorithm is a rather complex two-step process based on the use of a grid.

D. Other Fusion Procedures

There are other fusion procedures that resort to raw observations (i.e., points), do some operations on them, and build new line segments fitting the newly obtained points. Thus, differently from most of the previous ones, these methods are applicable only when line segments are obtained starting from sensors that perceive points. For example, in [3], line segments are used to approximate clusters of raw points acquired by sensors within a SLAM framework. When two clusters are considered to correspond, they are merged in a new cluster whose parameters are calculated using a variant of the Incremental Principal Component Analysis algorithm. Another example is the method used in [4], where the authors propose to evaluate whether two polylines can be

fused on the basis of the similarity of their convex arcs. The fusion of the two polylines is performed simulating a laser range scanner, thus reducing the problem to the fusion of points. The solution proposed in [16] describes a line segment using the center of gravity of the points it approximates and the direction θ of its supporting line. Line segments are grouped in clusters and, for each cluster, a new line segment is generated. Its parameters are the weighted average of the parameters of the line segments in the cluster, where the weights are the variances of the centers of gravity. Finally, in [17], the authors propose a procedure to merge a newly acquired line segment s and an old line segment s'. The procedure finds a line segment that approximates the measured points that form s and the points obtained by sampling s'.

IV. EXPERIMENTAL ANALYSIS

In this section, we experimentally analyze some of the methods that have been proposed for reducing the size of maps by eliminating redundant line segments. We first introduce the methods we selected for comparison and, then, we present the experimental results we obtained by applying these methods to different data sets.

A. The Compared Methods

We selected a representative method for each one of the following classes defined in Section III: basic fusion procedures, fusion procedures for pairs of line segments, and fusion procedures for sets of line segments. Our selection of representative methods from the above classes has been biased by the availability of source code and by the simplicity of implementation. This resulted in using two methods we developed in previous works and in implementing a simple method from scratch. In our opinion, this sample covers the spectrum of the current methods for reducing the number of line segments in maps. Due to space limitations, in the following we only describe the basic features of the methods we compare. Please refer to the original papers for details.

1) Fusing Consecutive Line Segments (FC): This fusion method is very basic. It fuses together line segments that are consecutive or close and aligned (similarly to [11]): typically, it applies to line segments representing long walls (e.g., a hallway). In order to fuse two line segments s and s' two fusion conditions must be satisfied:

(1) the angle between s and s' is smaller than a given threshold T_θ: $\widehat{ss'} < T_\theta$;
(2) the Euclidean distance between an extreme point of s and an extreme point of s' is smaller than a given threshold T_D.

If two line segments satisfy both these conditions, then they are replaced by a third line segment that connects the pair of farthest extreme points of the initial line segments.

2) Fusing Pairs of Line Segments (FP): This method has been presented in [10] and is based on the idea of iteratively fusing two line segments in a new line segment, namely for this method $|L| = 2$ and $|\hat{L}| = 1$.

A line segment s is represented by the following tuple: $s = \langle \theta, \rho, (x_1, y_1), (x_2, y_2), \sigma_\theta, \sigma_\rho \rangle$, where θ and ρ are the parameters (angle and distance from origin, respectively) of the line supporting s, (x_1, y_1) and (x_2, y_2) are the coordinates of the extreme points of s, and σ_θ and σ_ρ represent the uncertainty of θ and ρ, respectively.

The fusion conditions for deciding whether two line segments s and s', coming from the scan S^t and the current map M^t, respectively, can be fused are:

(1) the angle between s and s' is smaller than T_θ: $\widehat{ss'} < T_\theta$,
(2) the distance between s and s' is smaller than T_D: $D(s, s') < T_D$,
(3) the projections s_p and s'_p of s and s' on their bisectrix (i.e., on the line that bisects the smaller angle formed by s and s') overlap: $s_p \cap s'_p \neq \emptyset$.

The distance between s and s' is calculated as $D(s, s') = \min(d(s, s'), d(s', s))$, where $d(s, s') = \max_{p \in s}(\min_{p' \in s'}(||p - p'||))$, and $||\cdot||$ is the Euclidean distance. Note also that the thresholds T_θ and T_D are the main parameters of both the previous and this method. At each iteration of the fusion procedure, the set L is composed of two line segments s and s' that satisfy the three above conditions.

Given the line segments $s = \langle \theta, \rho, (x_1, y_1), (x_2, y_2), \sigma_\theta, \sigma_\rho \rangle$ and $s' = \langle \theta', \rho', (x'_1, y'_1), (x'_2, y'_2), \sigma'_\theta, \sigma'_\rho \rangle$ in L, the fusion algorithm for calculating the resulting line segment $\hat{s} = \langle \hat{\theta}, \hat{\rho}, (\hat{x}_1, \hat{y}_1), (\hat{x}_2, \hat{y}_2), \hat{\sigma}_\theta, \hat{\sigma}_\rho \rangle$ runs in the following way. The parameters of the line supporting $\hat{s}$ are:

$$\hat{\theta} = \frac{\omega_\theta \cdot \theta + \omega'_\theta \cdot \theta'}{\omega_\theta + \omega'_\theta} \qquad \hat{\rho} = \frac{\omega_\rho \cdot \rho + \omega'_\rho \cdot \rho'}{\omega_\rho + \omega'_\rho}$$

where the weights ω_θ and ω_ρ are defined as:

$$\omega_\theta = \frac{l}{l + l'} \cdot \frac{1}{\sigma_\theta^2} \qquad \omega_\rho = \frac{l}{l + l'} \cdot \frac{1}{\sigma_\rho^2}$$

where l and l' are the length of s and s', respectively. In a similar way, ω'_θ and ω'_ρ can be defined. The weights of a line segment are directly proportional to its length and inversely proportional to its uncertainty. This reflects the idea that long line segments are more "reliable" than short line segments.

The extreme points $(\hat{x}_1, \hat{y}_1)$ and $(\hat{x}_2, \hat{y}_2)$ of $\hat{s}$ are calculated by projecting the extreme points of s and s' on the line with parameters $\hat{\theta}$ and $\hat{\rho}$ and by selecting, among the four projected points, the pair of farthest points.

Finally, the uncertainty $\hat{\sigma}_x$ (either $\hat{\sigma}_\theta$ or $\hat{\sigma}_\rho$) of $\hat{s}$ is calculated as follows. When $l \gg l'$, $\hat{\sigma}_x$ is equal to σ_x (s "absorbs" s'); conversely, when $l \ll l'$, $\hat{\sigma}_x$ is equal to σ'_x (s is "absorbed" by s'). When $l = l'$, $\hat{\sigma}_x$ is calculated as:

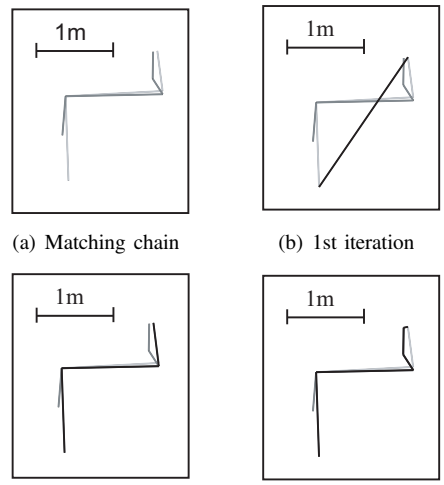
$$\hat{\sigma}_x = \sqrt{\left[\frac{1}{\sigma_x^2} + \frac{1}{\sigma_x'^2}\right]^{-1}}$$

In other cases, the value for $\hat{\sigma}_x$ is calculated as a linear combination of the previous values.

3) Fusing Sets of Line Segments (FS): This method has been presented in [9] and is based on the idea of fusing a set of line segments in a new polyline, namely, in a new set of line segments. Thus, for this method $|L| \geq 2$ and $|\hat{L}| \geq 1$.

The fusing conditions in this case are as follows. We say that two line segments match when they have a positive matching value. The *matching value* between two line segments s and s' (belonging to the new scan S^t and to the current map M^t, respectively) is calculated as follows. We project s' on the line supporting s thus obtaining a projected line segment s'_p and then we compute the length l_c of the common part of s and s'_p; we do the same but projecting s on s', obtaining l'_c. The matching value of s and s' is calculated as the average of l_c and l'_c. When s and s' do not intersect, the matching value is multiplied by $0.95^{D(s,s')/T_P}$ to penalize the match between line segments that are far away. Note that 0.95 is an empirical constant whose value has been determined during experimental activities, $D(s, s')$ is the distance between two line segments, calculated as above, and T_P is the threshold under which two points are considered to coincide (T_P has been set to $10\,\mathrm{mm}$ or to $15\,\mathrm{mm}$ in our experiments). When two line segments have a positive matching value they (supposedly) represent the same part of the environment. The matching value of s and s' is 0 when they are too far away, namely when $D(s, s') > 50\,\mathrm{mm}$. The matching value is set to 0 also when the two line segments intersect and are longer than a threshold (usually set to $80\,\mathrm{cm}$ or $100\,\mathrm{cm}$ in our experiments).

The fusion algorithm for this method is based on the notion of matching chain. A *matching chain* is a set of pairs of line segments that have a positive matching value. Given a pair, the first line segment is from the new scan and the second line segment is from the current map. Formally, a matching chain can be represented as $C = \{\langle s, s'\rangle | s \in S^t$ and $s' \in M^t$ have a positive matching value$\}$ such that if $\langle s, s'\rangle \in C$, then all the line segments $\bar{s}$ that have a positive matching value (namely, have matched) with s or s' belong to C; i.e., $\langle s, \bar{s}\rangle \in C$ and $\langle \bar{s}, s'\rangle \in C$. Each matching chain (i.e., each set of pairs of corresponding line segments) is fused in a single polyline. This polyline is built by iteratively building a sequence of approximating polylines $P_0, P_1, \ldots$ that converges to the polyline P that adequately approximates (and substitutes in the updated map) the line segments in C. The polyline P_0 is composed of a single line segment connecting the pair of farthest points (extremes of the line segments) in C. Given the polyline P_{n-1}, call s the line segment in (a pair belonging to) C that is at the maximum distance from its (closest) corresponding line segment $\bar{s}$ in P_{n-1}. If the distance $D(s, \bar{s})$ is less than the acceptable error threshold (set to $15\,\mathrm{mm}$ in our experiments), then P_{n-1} is the final approximation P. Otherwise, s is inserted in P_{n-1} to substitute $\bar{s}$ and s is connected to the two closest line segments in P_{n-1} to obtain the new polyline P_n. A line segment s in (a pair in) C is considered only once for insertion in the polyline. An example of this fusion algorithm is shown in Fig. 1.



(a) Matching chain (b) 1st iteration

(c) 2nd iteration (d) 3rd iteration

Fig. 1: Iterative construction of an approximating polyline (in black) for a matching chain (in two levels of gray)

B. Experimental Results

The experimental evaluation of the selected methods for fusing line segments has been done both on data collected in our laboratory and on publicly available data.

In our laboratory, we used a SICK LMS 200 laser range scanner to acquire a sequence of scans (distance measurements) forming a loop about $40\,\mathrm{m}$ long. The scans were acquired in different environments containing hallways with rectilinear walls, open spaces, and scattered obstacles. This Polimi data set is composed of a sequence of 28 scans. We also applied the methods to the stanford-gates1 data set available online at the Robotics Data Set Repository (Radish) [18]. This data set is a 30-minute tour through the first floor of the Stanford's Gates Computer Science building. We extracted a sequence of 59 laser scans acquired in a portion of the environment and taken $4\,\mathrm{s}$ apart.

In both cases, the raw points in a scan have been approximated by line segments, following the method of [11]. We then applied the registration method of [9] in order to determine the rototranslation that aligns a scan to the next one in the sequences. Once a scan had been aligned to the current map, the different fusion methods of Section IV-A have been applied to obtain the updated map. We performed both a quantitative and a qualitative analysis. Since we are interested in the performance of methods for reducing redundant line segments, we look at issues like the number of line segments and their characteristics (e.g., their length) and not at the correspondence between the map and ground truth data, being this last issue mainly related to the acquisition and registration methods used (steps (a) and (b) of Section II). In our experiments, we qualitatively noted that, if the scan and the map have been correctly aligned, the fusion process usually does not affect the correctness of the map w.r.t. the ground truth as it simply fuses together redundant line segments representing the same object, without altering

the "form" of the map. In this perspective, maps with a small number of long line segments are considered more desirable than maps with a large number of short line segments.

Table I shows some statistics about the final maps for the two data sets obtained by simply aggregating the line segments of the scans (agg, in the first row) and by applying the methods of Section IV-A (in the other three rows). The table reports the following quantities: the number of line segments, the average length $\bar{l}$ of line segments, and the standard deviation σ_l. Note that $\bar{l}$ and σ_l give an idea about the size of the obstacles in the map and their distribution. All the fusion methods do much better than the upper bound represented by agg. That said, FS generates the smallest final maps. This is more evident in the Stanford data set, that is larger than the Polimi data set. Moreover, we note that σ_l for the map generated with FS is relatively small. This suggests that the line segments in the final map are more "uniform" than those in the final map generated with the other methods. In particular, by visually inspecting the final maps, we noted that there are less short line segments. The fact that FS generates maps composed of fewer longer line segments does not imply that these line segments represent better the environment, because, as said, the precision of the map depends mainly on the acquisition and registration methods used.

Since FC and FP are based on the same parameters T_θ and T_D (recall Section IV-A), we compared their behavior as parameter values change. Not surprisingly, looser threshold values allow more line segments to be fused together: thus the final maps have less segments and their average length tends to increase (see Fig. 2). Moreover, we noted that looser thresholds affect the precision of the final map because of the larger approximation errors of the fusion procedure w.r.t. the acquisition and registration methods employed. These results suggest that a general method for fusing line segments should not influence the precision of the map (which mainly depends on the acquisition and registration methods) in attempting to reduce its size (number of line segments). Finding values of parameters that guarantee a good balance between these two aspects depends both from the environment mapped and from the application for which the map is built for.

We also qualitatively compared the final maps obtained. For example, Fig. 3 shows the final maps for the Polimi data set obtained with the three fusion methods (with $T_\theta = 20°$ and $T_D = 300\,\mathrm{mm}$). The map obtained with FC shows many overlapping or intersecting line segments that are not correctly fused. The method FC often fails in fusing both long and short line segments and thus the final map contains overlapping line segments even in the hallway where many long line segments could be easily fused. The method FP, on the other hand, performed well in fusing long line segments, but it often failed in fusing short line segments or chains of line segments. Fig. 4 shows a detail of the same portion of the final maps. The chain of line segments has been correctly fused by the method FS (e.g., the bunch of line segments on the bottom left of the map, that represent a wall, has been correctly replaced with a single long line segment); the method FP failed in fusing most of the line segments and created many intersecting and overlapping line segments representing the same world feature. In this portion, the method FC performed better than FP but left many overlapping line segments as well.

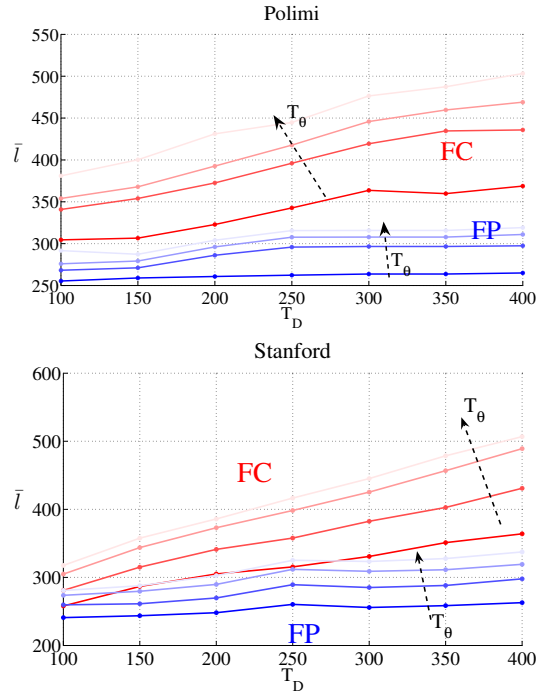


Fig. 2: Average length $\bar{l}$ of line segments for FC and FP with different threshold values (the lighter the color the larger T_θ)

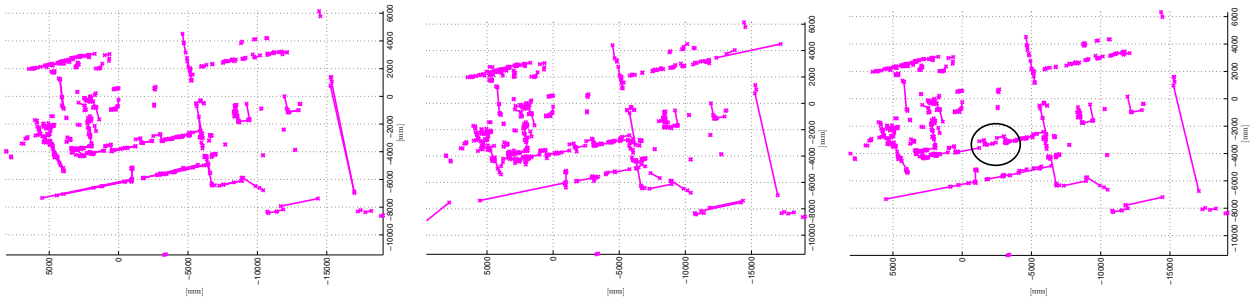
	POLIMI			STANFORD		
	# Seg.	$\bar{l}$	σ_l	# Seg.	$\bar{l}$	σ_l
agg	658	328.02	694.96	1507	268.27	321.09
FC	388	476.45	878.86	789	445.23	604.12
FP	365	315.77	699.33	765	323.44	573.77
FS	358	262.15	560.79	673	275.14	398.54

TABLE I: Statistics for the final maps for the two data sets (lengths are in mm)

V. CONCLUSIONS AND FUTURE WORKS

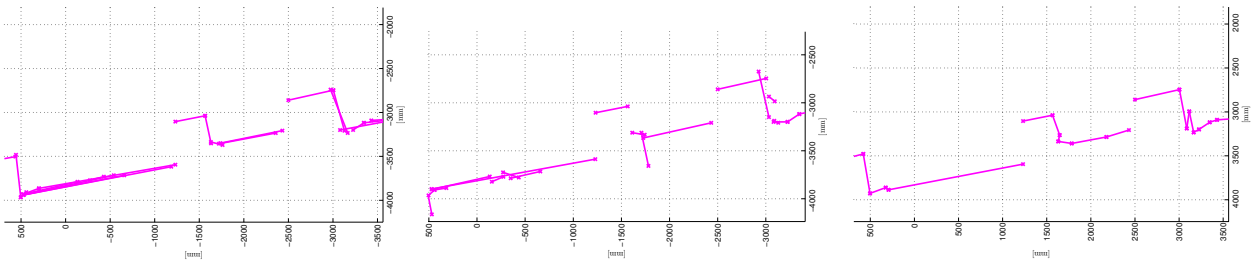
In this paper, we analyzed some methods for fusing redundant line segments in maps. The sample of methods we considered are representative of those proposed in literature, according to the taxonomy of Section III. Summarizing our results, we attempt to set out some guidelines for the development of a general method for reducing line segments in maps.

- Fusing redundant line segments is a complex problem for which basic solutions do not usually work well. For example, experiments show that the FC and the FP methods are not able to cope with many situations and leave several redundant line segments in the maps.
- The definition of a general approach for reducing the size of segment-based maps should consider the fusion of *sets* of line segments (as the FS method) and should



(a) FC (b) FP (c) FS

Fig. 3: Final maps



(a) FC (b) FP (c) FS

Fig. 4: Detail of the area circled in Fig. 3 (c) for the three maps

preserve the precision of the map while reducing its size.

- One of the hardest problems is the definition of fusion conditions able to capture when two or more line segments represent the same object in the environment.

Future work will address the definition and analysis of new fusion methods starting from the results presented here. In doing so, it could be interesting to apply the general method of [19] for approximating line arrangements with fewer lines to the specific application considered in this paper, namely the reduction of the number of redundant line segments in a map. Moreover, it would be interesting to perform further experiments in outdoor and 3D environments and with SLAM-based mapping methods.

VI. ACKNOWLEDGMENTS

The authors would like to thank Brian Gerkey for providing the Stanford data set and Igor Giussani for some experimental results.

REFERENCES

[1] S. Thrun, "Robotic mapping: A survey," in *Exploring Artificial Intelligence in the New Millenium*, G. Lakemeyer and B. Nebel, Eds. Morgan Kaufmann, 2003, pp. 1–35.

[2] D. Austin and B. McCarragher, "Geometric constraint identification and mapping for mobile robots," *ROBOT AUTON SYST*, vol. 35, pp. 59–76, 2001.

[3] E. Brunskill and N. Roy, "SLAM using incremental probabilistic PCA and dimensionality reduction," in *Proc. ICRA*, 2005, pp. 342–347.

[4] L. J. Latecki, R. Lakaemper, X. Sun, and D. Wolter, "Building polygonal maps from laser range data," in *Proc. CogRob*, 2004, pp. 56–62.

[5] E. Zalama, G. Candela, J. Gómez, and S. Thrun, "Concurrent mapping and localization for mobile robots with segmented local maps," in *Proc. IROS*, 2002, pp. 546–551.

[6] M. López-Sánchez, F. Esteva, R. L. de Màntaras, C. Sierra, and J. Amat, "Map generation by cooperative low-cost robots in structured unknown environments," *AUTON ROBOT*, vol. 5, pp. 53–61, 1998.

[7] V. Nguyen, S. Gächter, A. Martinelli, N. Tomatis, and R. Siegwart, "A comparison of line extraction algorithms using 2d range data for indoor mobile robotics," *AUTON ROBOT*, vol. 23, no. 2, pp. 97–111, 2007.

[8] F. Lu and E. Milios, "Globally consistent range scan alignment for environment mapping," *AUTON ROBOT*, vol. 4, no. 4, pp. 333–349, 1997.

[9] F. Amigoni, S. Gasparini, and M. Gini, "Building segment-based maps without pose information," *P IEEE*, vol. 94, no. 7, pp. 1340–1359, 2006.

[10] F. Amigoni, G. Fontana, and F. Garigiola, "A method for building small-size segment-based maps," in *Proc. DARS*, 2006, pp. 11–20.

[11] H. H. Gonzáles-Baños and J. C. Latombe, "Navigation strategies for exploring indoor environments," *INT J ROBOT RES*, vol. 21, no. 10-11, pp. 829–848, 2002.

[12] A. Garulli, A. Giannitrapani, A. Rossi, and A. Vicino, "Simultaneous localization and map building using linear features," in *Proc. ECMR*, 2005.

[13] J. Gonzalez, A. Ollero, and A. Reina, "Map building for a mobile robot equipped with a 2D laser rangefinder," in *Proc. ICRA*, 1994, pp. 1904–1909.

[14] S. Pfister, S. Roumeliotis, and J. Burdick, "Weighted line fitting algorithms for mobile robot map building and efficient data representation," in *Proc. ICRA*, 2003, pp. 1304–1311.

[15] Y. L. Ip, A. B. Rad, K. M. Chow, and Y. K. Wong, "Segment-based map building using enhanced adaptive fuzzy clustering algorithm for mobile robot applications," *J INTELL ROBOT SYST*, vol. 35, pp. 221–245, 2002.

[16] L. Zhang and B. Ghosh, "Line segment based map building and localization using 2D laser rangefinder," in *Proc. ICRA*, 2000, pp. 2538–2543.

[17] R. Mazl and L. Preucil, "Building a 2D environment map from laser range-finder data," in *Proc. IV*, 2000, pp. 290–295.

[18] A. Howard and N. Roy, "The robotics data set repository (radish)," http://www.robot.uji.es/benchmarks/index.html.

[19] D. Dobkin and A. Tal, "Efficient and small representation of line arrangements with applications," in *Proc. Symposium Computational Geometry*, 2001, pp. 293–301.