Change detection in 3D environments based on Gaussian Mixture Model and Robust Structural Matching for Autonomous robotic applications

P. Núñez, P. Drews Jr, A. Bandera, R. Rocha, M. Campos and J. Dias

Abstract— **The ability to detect perceptions which were never experienced before, *i.e.* novelty detection, is an important component of autonomous robots working in real environments. It is achieved by comparing current data provided by its sensors with a previously known map of the environment. This often constitutes an extremely challenging task due to the large amounts of data that must be compared in real-time. With respect to previously proposed approaches, this paper detects changes in 3D environment based on probabilistic models, the Gaussian Mixture Model, and a fast and robust combined constraint matching algorithm. The matching allows to represent the scene view as a graph which emerges from the comparison between Mixtures of Gaussians. Finding the largest set of mutually consistent matches is equivalent to find the maximum clique on a graph. The proposed approach has been tested for mobile robotics purposes in real environments and compared to other matching algorithms. Experimental results demonstrate the performance of the proposal.**

I. Introduction

In order to autonomously explore and navigate on an unknown and dynamic environment, mobile robots typically require to determine their pose (position and orientation) and to simultaneously build a map of this environment based on perceptual data. In this situation, the ability to detect and respond suitably to scene changes arises as an useful component. For instance, in robotic surveillance and security systems [1], environment changes may be risky situations requiring the activation of some kind of alarms with which the robot should be aware of. In a similar way, robots exploring dangerous environments (*e.g.* abandoned mines [2]), should solve and warn about risk situations when a change is detected along its motion with respect to the known map. Then, the novelty detection arises as a mechanism which allows the robot to adapt itself to new situations and to continue its operation, updating the knowledge of the environment and focusing the attention on a specific region of interest [3].

The basic idea behind most current novelty detection approaches in mobile robotics is that the robot carries sensors to perceive the environment and to match the obtained data with the expected data available in the map. The success of this matching process is conditioned on (1) the existence of accurate sensors capable of obtaining raw information from the environment, (2) the availability of fast and reliable algorithms capable of extracting a high-level representation from the large sets of noisy and uncertain data, and (3) the existence of an accurate method that is be able to detect the change according to the employed representations.

With respect to the first question, 3D laser range sensors or vision-based systems can be used. Applying vision to feature extraction leads to increase CPU usage due to the complexity of the algorithms required. Conversely, a 3D laser range scanner is capable of collecting such high quality range data but it suffers from very small number of specular reflections. The angular uncertainty of the laser sensor is very small and, therefore, it can provide to the robot a very fine description of the surroundings. For the second issue, pattern recognition and image analysis background have inspired different methods for clustering of 3D points. Thus, simple methods have been broadly used to support mobile robot operation extracting planar structures or more compact models [4], [5], [3]. Other possibility is to address the 3D clustering problem within the framework of statistical approaches, *e.g.* using Mixture Models [6] or Principal Component Analysis [7]. Specifically, Mixtures of Gaussian distributions provide good models of point clusters, as it was demonstrated in previous works by the authors (see [8] and [9]). Finally, with respect to the third question, several metrics have been proposed to detect changes using the data acquired by the sensors. Typically, the aim is to compare the clouds of 3D points associated to each dataset and detect those pairs of points whose distance is higher than a fixed threshold. In order to reduce the computational cost of this process, more complex metrics, which include statistical information associated to the underlying point distributions, have been used. In Tomasi *et al.* 's work [10], the Earth Mover's Distance was proposed as a new metric for solving this kind of situation. This metric was employed in a previous work by the authors where a greedy algorithm is used for detecting changes in the robot environment [9]. The main disadvantage of this approach is its strong dependence on the number of Gaussians associated to the map, which implies an increasing error in the number of detected changes.

The proposed approach consists of a novel algorithm for detecting changes in the environment of the robot using the 3D data acquired by laser range sensors. Fig. 1 shows an overview of the method. Firstly, the environment information is simplified through a multi-scale sampling technique in order to reduce the computation burden of detecting changes.

This work has been partially supported by the PROMETHEUS, EU-FP7-ICT-2007-1-214901 project, by MICINN Project n. TIN2008-06196 and TSI-020301-2009-2, CAPES, CNPQ and FEDER funds.

P. Núñez is member of the ISIS Group, Universidad de Málaga, and Robolab Group, Dept. Tecnología de los Computadores y las Comunicaciones, Universidad de Extremadura, Spain.

Paulo Drews Jr. and M. Campos are with Dept. Computer Science, Federal University of Minas Gerais, Brazil. (paulol@dcc.ufmg.br)

A. Bandera is with ISIS Group, Universidad de Málaga, Spain

Rest of authors are with the Institute of Systems and Robotics, Dept. Electrical and Computer Engineering, University of Coimbra, Portugal.

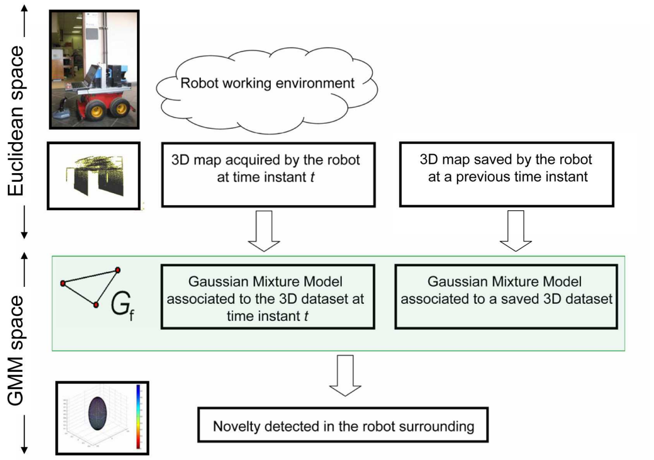
Euclidean space
GMM space
Robot working environment
3D map acquired by the robot at time instant t
3D map saved by the robot at a previous time instant
G_f
Gaussian Mixture Model associated to the 3D dataset at time instant t
Gaussian Mixture Model associated to a saved 3D dataset
Novelty detected in the robot surrounding

Fig. 1. Problem statement: given the 3D information acquired by the laser sensor at time instant t and a known map on the environment, the robot detects changes in the scene using a structural matching algorithm.

Next, the data to be compared is compacted according to the well-known Gaussian Mixture Model (GMM). The GMM assumes that the probability density function (pdf) of the cloud of points can be modelled by a mixture of Gaussian distributions [8]. Finally, the system performs a structural matching stage in the GMM feature space in order to find the novelty in the scene.

The latter part of the process is the main novelty introduced in the method by this paper. Most of the matching algorithms in the literature are limited by the independence assumption, where each possible association is considered as a separate problem, with no influence on the association of other possible associations located in the same vicinity. The proposed work uses a robust and fast algorithm which does not only take into account the Gaussians distributions similarity to define the global matching (absolute constraints), but also relative constraints related to local structural information. These two constraints, both absolute and relative, are used to compute a consistency matrix for all pairwise matching combinations. This matrix is used to find the largest set of mutually consistent matchings. This process is equivalent to find the maximum clique on a graph defined by this adjacency matrix.

The rest of the paper is organized as follows. Sec. II briefly reviews the maximum clique problem and the branch-and-bound algorithm which will be employed to solve it in this proposal. Sec. III describes the proposed novelty detection approach. Experimental results are shown in Sec. IV. This Section also includes a comparison of the proposal with other related approaches. Finally, Sec. V draws the main conclusions and future work.

II. Maximum Clique Problem

Let $G = (N, E)$ be an undirected graph with node set $N = \{n_1, ...n_n\}$. Two nodes n_i and n_j are said to be adjacent if they are connected by an edge $e_{ij} \in E$. A clique of a graph is a set of nodes where all of them are adjacent, and a maximum clique is the largest among all cliques in a graph. The problem of finding the maximum clique problem is computationally equivalent to some other important graph problems, *e.g.* the maximum independent (or stable) set problem and the minimum node cover problem. Since these are NP-hard problems, no polynomial time algorithms are expected to be found.

TABLE I

Fast Maximum clique algorithm [11]

```
function clique($U$, $size$)
1: if $|U| = 0$ then
2:       if $size > max$ then
3:             $max := size$
4:             New $record$; save it.
5:             $found := true$
6:       end if
7:       return
8: end if
9: while $U \neq \emptyset$ do
10:        if $size + |U| \leq max$ then
11:              return
12:        end if
13:        $i := min\{j|n_j \in U\}$
14:        if $size + c[i] \leq max$ then
15:              return
16:        end if
17:        $U := U \backslash \{n_i\}$
18:        clique($U \cup N(n_i)$; $size$ + 1)
19:        if $found = true$ then
20:              return
21:        end if
22: end while
23: return
function new
24: $max := 0$
25: for $i := n$ downto 1 do
26:        $found := false$
27:        clique($Si \cap N(n_i)$, 1)
28:        $c[i] := max$
29: end for
30: return
```

In this paper, the branch-and-bound fast algorithm proposed in [11] for the maximum clique problem is employed. Let $\{n\}_{i=1}^{n}$ be the set of nodes of the graph G and S_i be the subset $\{n_i, n_{i+1}, ...n_n\}$. Firstly, the maximum clique algorithm looks for cliques in S_n that contain n_n (the largest clique is $\{n_n\}$), then cliques in S_{n-1} that contain n_{n-1}, and so on. The algorithm is presented in Table I. The set of nodes adjacent to a node n_i is denoted by $N(n_i)$ and the number of nodes in the graph is n. The global variable max gives the size of a maximum clique when the algorithm terminates. The function $c(i)$ gives the largest clique in S_i. Obviously, for any $1 \leq i \leq n-1$, we have that $c(i) = c(i+1)$ or $c(i) = c(i+1)+1$. Moreover, we have $c(i) = c(i+1)+1$ iff there is a clique in S_i of size $c(i+1)+1$ that includes the node n_i. Therefore, starting from $c(n) = 1$, we search for such cliques. If a clique is found, $c(i) = c(i+1)+1$, otherwise $c(i) = c(i+1)$. The size of a maximum clique is given by $c(1)$. Old values of the function $c(i)$ enables the new pruning strategy (in line 14). That is, if we search for a clique of size greater than s, then we can prune the search if we consider n_i to become the $(j+1)$-th node and $j + c(i) \leq s$.

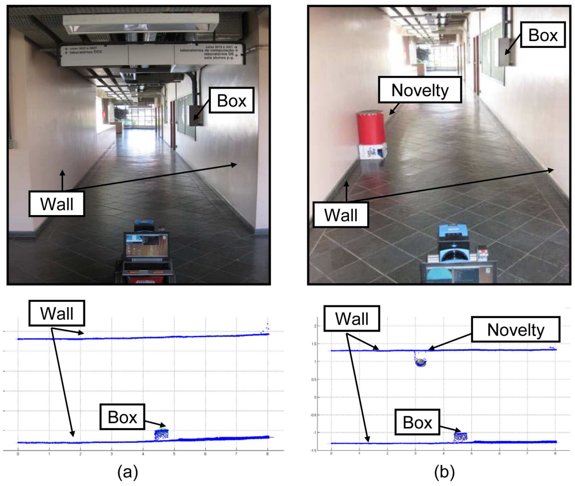

Fig. 2. The main goal of the proposed algorithm is to detect any novelty in the working environment of the robot (*e.g.* the cylinder in b).

III. Novelty detection in 3D environments

The main goal of novelty detection is *to determine any previously unknown feature* [3]. This section describes the proposed algorithm for detecting changes in the robot surrounding (*e.g.* novelty marked in Fig. 2). The proposed method is based on our previous works [8] and [9]. In the current approach, the 3D laser range data is preprocessed in order to reduce the number of points. Then, the comprised data is transformed from the Euclidean space to the GMM space. Finally, the novelty is detected using a structural matching algorithm. The main advantages of this approach are (i) its feasibility, due to the data simplification and posterior compression using GMM, and (ii) robust matching, due to the outliers removal and the use of the combined constraint matching method. A more detailed description of the algorithm is provided by the next subsections.

A. Pre-processing stage

The main aim of the pre-processing stage is to reduce the high density of points acquired by a typical 3D laser scanner. Specifically, the approach used in this work is based on the method proposed by Pauly *et al.* [12]. This method has one important contribution: it reduces the computation time while minimizing the losing of geometric information. Basically, it computes a multi-scale points cloud using binary space partition. The use of covariance analysis allows to compute the surface variation (σ) based on eigenvalues. Thereby, the points cluster P is split if its size, $|P|$, is larger than a given value and the surface variation is above a maximum threshold σ_{max}. The value of σ_{max} is set to 0.1, where the range of σ is $[0; \frac{1}{3}]$. This threshold has been empirically selected to a typical laser data density value.

This hierarchical cluster simplification builds a binary tree based on the split of each region. The split plane is defined by the centroid of P and the eigenvector associated to the greater eigenvalue (λ_2). Thus, the points cloud is always split along the direction of greatest variation. The multi-scale representation is based on the restriction level imposed to the tree. The tree grows until the cluster is just one point. The scale is chosen by setting values to the size of P, $|P|$, and to the σ_{max} value.

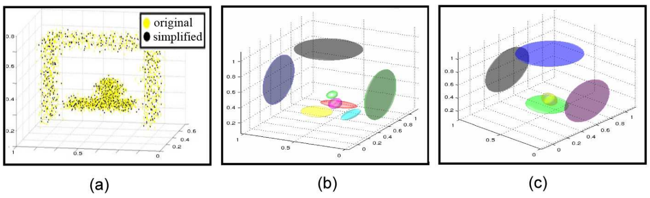

Fig. 3. a) 3-dimensional laser range data and comprised one (yellow and black, respectively); b) GMM associated to the original laser data; and c) GMM associated to the comprised laser data.

On the other hand, considering a points cloud obtained by a laser scanner, the ground plane is almost always present in the data. In this work, a simple method using RANSAC is used to fit a ground plane [13]. Finally, sparse outliers in the 3D scan laser data are removed based on the technique described in [14].

B. Gaussian Mixture Model (GMM)

A *Gaussian mixture model* (GMM) is a probability model for density estimation using a convex linear combination of Gaussians density functions. The GMM has the form:

$$f(\mathbf{x}, \Theta) = \sum_{k=1}^{K} p_k \, g(\mathbf{x}; \mu_k, \Sigma_k) \quad \left(\mathbf{x} \in \mathbb{R}^N\right) \tag{1}$$

In this model, each Gaussian is defined by a coefficient $p_k \geq 0$, which satisfies $\sum_{k=1}^{K} p_k = 1$, and by its mean and covariance matrix (μ_k and Σ_k). The GMM provides good models of clusters of points: each cluster corresponds to a Gaussian density with mean somewhere within the centroid of the cluster, and with a covariance matrix somehow measuring the spread of that cluster.

Given a set of points, it is possible to find the GMM Θ using the Expectation-maximization (EM) algorithm [8]. The size of K is selected using K_{max} and the MDL penalty function [15]. Fig. 3 illustrates the results of the preprocessing and GMM stages. In Fig. 3a, the 3D laser data and the comprised data provided by the pre-processing stage are drawn in yellow and black, respectively. The number of points has been reduced about 70%. Figs. 3b and 3c show the GMM associated to the original laser range data and to the comprised data, respectively. See [8] for further details.

C. Combined Constraint matching algorithm

In this section, the matching problem is formulated as a graph-theoretic data association problem. Thus, the fundamental data structure of this step is the correspondence graph [16], which represents valid associations between the two mixture of Gaussians (see Fig. 4). Cliques within the graph indicate mutual associations compatibility and, by performing a maximum clique search, the joint compatible association set emanated from the better matchings of Mixtures of Gaussians may be found. The construction of the correspondence graph is performed through the application of both relative and absolute constraints. Thus, nodes of the graph indicate individual association compatibility and they are determined by absolute constraints. On the other hand, the arcs of the correspondence graph indicate joint compatibility of the connected nodes. They are determined by relative constraints.

Let $\Theta = \{((\theta_1, p_1), \dots, (\theta_n, p_n))\}$ and $\Gamma = \{((\gamma_1, q_1), \dots, (\gamma_m, q_m))\}$ be the GMMs associated with two 3D scans, where $\theta_i(\mu_i, \Sigma_i)$ and $\gamma_j(\mu_j, \Sigma_j)$ are Gaussians functions, p_i and q_j are the weights associated to each Gaussian, and (μ_k, Σ_k) is a vector containing all the coordinates of the means μ_k and all the entries of the covariance matrix Σ_k. The method used to calculate the correspondence graph has two major steps:

1) *Definition of the nodes of the correspondence graph*. In the proposed method, graph nodes are associated to tentative matchings of Gaussian distributions from two GMMs, Θ and Γ, after applying an absolute constraint. Let $|\Theta| = n$ and $|\Gamma| = m$ be the number of Gaussians functions, respectively. Firstly, the algorithm generates the matrix T_t $(n \times m)$ for all pairwise combinations, by calculating the distance between the two Gaussian functions:

$$d_{\theta_i, \gamma_j} = max(d_{\mu_{ij}}, d_{\Sigma_{ij}}) \tag{2}$$

where $d_{\mu_{ij}}$ is the Euclidean distance between the two Gaussian functions using the coordinates of the mean vector, and $d_{\Sigma_{ij}}$ the distance between the covariance matrices associated to the Gaussian functions [17]. This is defined as:

$$d_{\Sigma_{ij}} = \sqrt{\sum_{k=1}^{N} ln^2 \lambda_k(\Sigma_i, \Sigma_j)} \tag{3}$$

where λ represents the generalized eigenvalues of Σ_i and Σ_j, and N is the dimensionality of the matrices. The matrix item associated to the matching of two similar Gaussian functions presents a low value. On the other hand, high values at T_t correspond to dissimilar features. Pairwise matched features whose matrix values are lower that a fixed threshold U_T^t constitute the set of tentative matchings. Thus, graph nodes are defined as the set of all possible combinations of these pairwise descriptors (*e.g.* node $(1_a,\ 1_b)$ in Fig. 4 is valid if θ_{1_a} is a possible correspondence of γ_{1_b}).

2) *Definition of the arcs of the correspondence graph*. For all pairwise combinations of matchings in T_t, a relative constraint matrix is calculated, R_t. To do that, a relative constraint on the space of the GMM is used. A pair of matched Gaussian functions (θ^i, γ^i) and (θ^j, γ^j) is consistent iff they satisfy the relative constraint:

$$max(\omega_{d_\mu}, \omega_{d_\Sigma}) \leq U_R^t, \tag{4}$$

being

$$\begin{aligned} \omega_{d_\mu} &= \sqrt{|(d_{\mu_{ij}}^{\Theta})^2 - (d_{\mu_{ij}}^{\Gamma})^2|} \\ \omega_{d_\Sigma} &= \sqrt{|(d_{\Sigma_{ij}}^{\Theta})^2 - (d_{\Sigma_{ij}}^{\Gamma})^2|} \end{aligned} \tag{5}$$

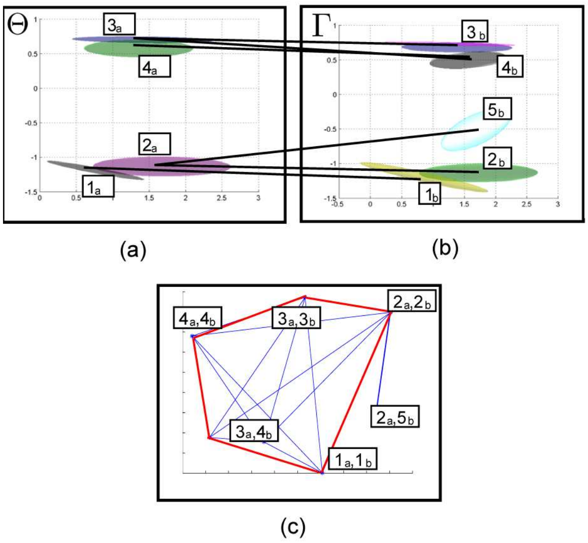

Fig. 4. Nodes represent tentative matchings when considered individually. Arcs indicate compatible associations, and a clique is a set of mutually consistent associations (*e.g.* the clique marked in red implies that the matching shown between a) and b) may coexist).

where U_R^t is a threshold defined by the user. Thus, the corresponding entry in the relative constraint matrix R_t contains a 1 value if the constraint is satisfied (arc in the graph), and 0 otherwise. For instance, in Fig. 4, the relative constraint between ($1a$, $1b$) and ($2a$, $2b$) matches, and then node ($1a$, $1b$) is connected to node ($2a$, $2b$). The relative constraint between ($2a$, $5b$) only matches with ($2a$, $2b$).

3) *Maximum clique detection and change description.* The set of mutually consistent matches which provides a largest clique is calculated. This is equivalent of finding the maximum clique on a graph with adjacency matrix R_t. The problem was briefly explained in Sec. II. After applying the maximum clique algorithm described in that Section, this step obtains a set of mutually compatible associations, *i.e.* a set of matched Gaussian functions (red lines in Fig. 4). In this way, the algorithm takes into account structural relationships to detect correct associations, which result in 3D points in the environment that are not associated to changes in the robot surrounding. Thus, the set of Gaussian functions in Θ which are not included in the clique represents the novelty detected by the algorithm. In Fig. 4, the only node which is not include in the clique, *i.e.* ($2a$, $5b$), is the novelty in the robot environment.

IV. Experimental results

In this section, the proposed change detection method has been analyzed in terms of robustness and computational load. The main novelty of this work, the combined constraint matching algorithm, which includes the search for the maximum clique on the graphs, is compared with other two matching approaches: (i) a simple matching algorithm based on the position of the Gaussian functions in the GMM space; and (ii) the greedy algorithm proposed by the authors in [9], which is based on Earth MoverŠs Distance (EMD). All the methods have been implemented in C++, and they are tested in a 1.66GHz Pentium PC computer with 1Gb of RAM.

A. *Change detection in real environment*

Novelty detection algorithm has been tested in different real environments inside the research area sited in the Minas Gerais Federal University, as is shown in Fig. 5. For the experiments drawn in Fig. 5, three different *novelties* were included in order to evaluate the results of the algorithm (a cylinder, a person and a box in Fig. 5a, respectively). Fig. 5b illustrates the 3D laser range data acquired by the robot after the simplification method. The GMM associated to the 3D map is shown in Fig. 5c, and the real novelty is marked in the figure. Results of the proposed algorithm are drawn in Fig. 5d. As it is shown in the figure, the novelty detection algorithm is able to extract the Gaussian functions associated to the changes in the environment.

B. *Evaluation of the robustness and time processing*

Robustness and computational load of the proposed matching algorithm have been evaluated and compared against two different matching methods: the greedy EMD-based algorithm [9], and a simple matching algorithm which is only based on an absolute constraint, *i.e.* the Euclidean distance between the mean vector associated to each Gaussian function. With the aim of validating the approach, the same set of 3D laser range data collected by the sensors has been used. For each 3D dataset, the Gaussian Mixture Model (Γ) is calculated using different numbers of Gaussian functions, ($m = \{10, 20, 30, 40\}$). The Gaussian functions associated to the novelty are manually selected (the total number is considered as *Total positives*). Next, each novelty detection algorithm is run and the number of Gaussian functions associated to correct and incorrect detected changes are manually counted, and they are considered as $NumberTrueNovelty$ and $NumberFalseNovelty$, respectively. In Figs. 6a-b, the same 3D dataset is segmented using different number of Gaussian functions (20 and 40, respectively). Fig. 6c represents the results of the EMD algorithm for the GMM shown in Fig. 6a. As it is shown in the figure, the novelty detected by the algorithm is incorrect. The novelty is correctly detected by the proposed algorithm in Fig. 6d.

With the aim of evaluating the robustness of the matching algorithm that is included in the proposed novelty detection algorithm, the following measurements are defined:

$$\begin{aligned} TruePos &= \frac{NumberTrueNovelty}{Totalpositives} \\ Error &= \frac{NumberFalseNovelty}{NumberFalseNovelty + NumberTrueNovelty} \end{aligned} \qquad (6)$$

The average performance of the matching methods after the total experiment is given in Fig. 6e-f, which represents the evolution of the $TruePos$ and $Error$ against the number of Gaussian functions used to define the GMM. From this figure, it can be noted that the average $TruePos$ value is high for each algorithm when the number of Gaussian functions is low ($m = 10$ in the figure). After this value, due to the high number of outliers, the efficiency of the algorithms decreases. However, it can be appreciated that the structure-based features matching algorithm used in this work presents a strong robustness to detect correct changes. Similar to the $TruePos$ value, the error rapidly increases for all matching algorithms analyzed in this comparative study, being this decreasing less pronounced in the proposed structure-based features matching algorithm. These two graphs show the high performance of the maximum clique strategy for solving novelty detection problems. Finally, Table II illustrates with details the time consumption of the algorithm for the experiments described in this section (m = 20). As is shown in the table, the algorithm performance is faster that the proposed in [9]. These results are similar when the number of Gaussians m is modified.

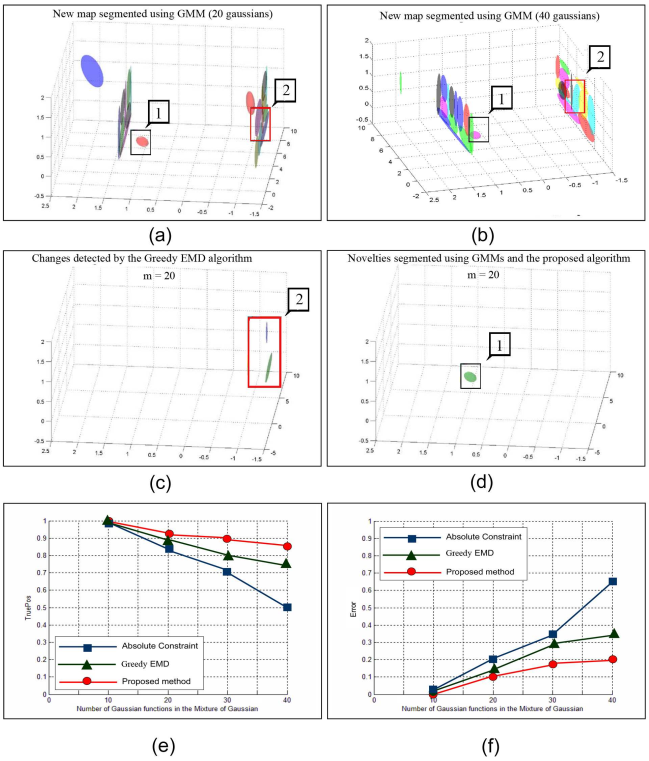

Fig. 6. Novelty detection algorithm: a-b) map the environment segmented using GMM (m = 20 and m = 40, respectively). Novelty is indicated by the label ’1’; c) wrong changes have been detected by the greedy EMD algorithm (label ’2’); d) The novelty detected by the proposed algorithm has been indicated by the label ’1’; e-f) TruePos and Error evolution according to the number of gaussian used to segment the point clouds.

V. Conclusion and Future work

This paper has presented a new method to directly detect changes in the environment of a robot using a 3-D laser range finder. Gaussian Mixture Model has been used to obtain a new representation of the point clouds and a novel structural matching algorithm is employed to quantify the existence of changes in the scene. The proposed method has

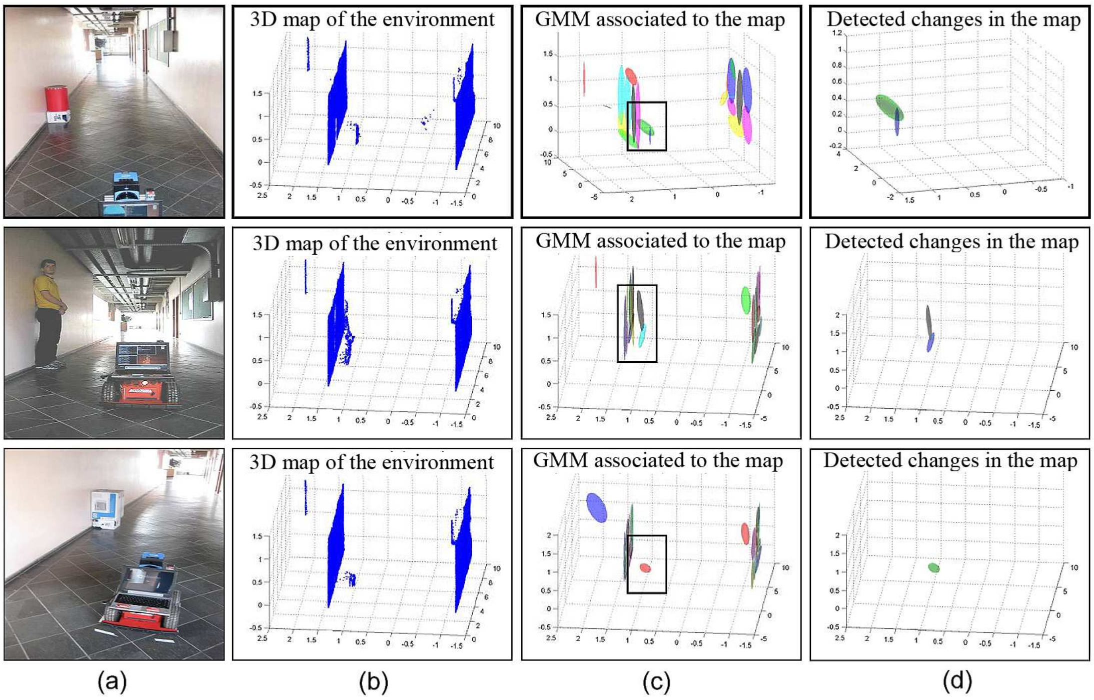

Fig. 5. Three different experiments where the novelty detection algorithm is evaluated. See text for further details.

TABLE II

COMPARATIVE STUDY OF DIFFERENT NOVELTY DETECTION ALGORITHMS.

		Number of Points		Time Elapsed (s)						
		Reference Map	Current Map	Simpl. Ref. Map	Simpl. Cur. Map	GMM Ref. Map	GMM Cur. Map	Greedy EMD	Absolute	Proposed
Real Data - Test Area 1	Simplified	21631	21744	0.43	0.43	176.84	164.13	0.020	0.012	0.014
	Complete	79171	79633	-	-	627.02	534.51	0.060	0.041	0.046
Real Data - Test Area 2	Simplified	21631	21744	0.43	0.46	176.91	110.4	0.020	0.011	0.014
	Complete	79171	81134	-	-	624.13	1342.71	0.03	0.023	0.027
Real Data - Test Area 3	Simplified	21631	21865	0.41	0.46	167.47	108.76	0.040	0.020	0.028
	Complete	79171	80112	-	-	625.23	865.12	0.051	0.036	0.041

been compared with two different matching methods in the mathematical space of Mixture of Gaussians. Experimental results in various real scenarios demonstrate the feasibility of the approach.

Future work will be focused on the integration of the current novelty detection algorithm into real robotic applications, like surveillance or exploration of dangerous environment, where detecting and segmenting novelties is important.

REFERENCES

[1] H. Andreasson, M. Magnusson, and A. Lilienthal. "Has something changed here? Autonomous Difference Detection for Security Patrol Robots". In *Proc. of IEEE/RSJ IROS*, pp. 3429-3435, 2007.

[2] A. Nuchter, H. Surmann, K. Lingemann, J. Hertzberg, and S. Thrun. "6D SLAM with an Application in Autonomous Mine Mapping". In *Proc. of IEEE International Conference on Robotics and Automation*, pp. 1998-2003, 2004.

[3] H. Vieria and U. Nehmzow. "Visual novelty detection with automatic scale selection", *Robotic and Autonomous Systems*, V. 55, No. 8, pp. 693-701, 2007.

[4] N. Lomiere. "A generic methodology for partitioning unorganised 3D point clouds for robotic vision". In *Proc. of the 1st Canadian Conference on Computer and Robot Vision*, pp. 64-71, 2004.

[5] S. Ioannis and L. Marius. "Automated feature-based range registration of urban scenes of large scale". In *Proc. of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 555–561, 2003.

[6] G. McLachlan and D. Peel. "Finite Mixture Models". Wiley, New-York, 2000.

[7] I. Jolliffe. Principal Component Analysis. Springer series. 2002.

[8] P. Núñez, P. Drews Jr, R. Rocha, M. Campos and J. Dias. "Novelty Detection and 3D Shape Retrieval based on Gaussian Mixture Models for Autonomous Surveillance Robotics". In *Proc. of IEEE/RSJ International Conference on Robots and Systems*, pp. 4724-4730, 2009.

[9] P. Drews-Jr, P. Núñez, R. Rocha, M. Campos and J. Dias. "Novelty Detection and 3D Shape Retrieval using Superquadrics and Multi Scale Sampling for Autonomous Mobile Robots". Accepted in *Proc. of IEEE International Conference on Robotics and Automation*, 2010.

[10] C. Tomasi, Y. Rubner and L. Guives. "A metric for distributions with applications to image databases", in *Proc. of ICCV*, pp. 59-66, 1998.

[11] P. Östergard, "A fast algorithm for the maximum clique problem", in *Discrete Applied Mathematics*, 120, pp. 197Ű-207, 2002.

[12] M. Pauly, M. Gross and L. Kobbelt, "Efficient simplification of point-sampled surfaces", in *Proc. of IEEE Visualization*, pp 163-170, 2002.

[13] K. Lai and D. Fox, "3D Laser Scan Classification Using Web Data and Domain Adaptation", *Robotics: Science and Systems (RSS)*, 2009.

[14] R. B. Rusu, Z. C. Marton, N. Blodow, M. Dolha and M. Beetz, "Towards 3D Point cloud based object maps for household environments", *Robotics and Autonomous Systems*, V. 56, No. 11, pp. 927-941, 2008.

[15] G. Rissanen, "Modeling the shortest data description", *Automatica*, V. 14, pp. 465-471, 1978.

[16] H. G. Barrow, R. M. Burstall, Subgraphs isomorphism, matching relational structures and maximal cliques, Information Processing Letters 4 (1976) 83-84.

[17] W. Forstner and B. Moonen, "A metric for covariance matrices", in Technical Report, Dpt. of Geodesy and Geoinformatics, University of Stuttgart, Germany, 1999.