The 2009 IEEE/RSJ International Conference on
Intelligent Robots and Systems
October 11-15, 2009 St. Louis, USA

Novelty Detection and 3D Shape Retrieval based on Gaussian Mixture Models for Autonomous Surveillance Robotics

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

Abstract— **This paper describes an efficient method for retrieving the 3-dimensional shape associated to novelties in the environment of an autonomous robot, which is equipped with a laser range finder. First, changes are detected over the point clouds using a combination of the *Gaussian Mixture Model* (GMM) and the *Earth Mover's Distance* (EMD) algorithms. Next, the shape retrieval is achieved using two different algorithms. First, new samplings are generated from each Gaussian function, followed by a *Random Sampling Consensus* (RANSAC) algorithm to retrieve geometric primitives. Furthermore, a new algorithm is developed to directly retrieve the shape according to the mathematical space of Gaussian mixture. In this paper, the set of geometric primitives has been limited to the set *C = {sphere, cylinder, plane}*. The two shape retrieval methods are compared in terms of computational cost and accuracy. Experimental results in various real and simulated scenarios demonstrate the feasibility of the approach.**

I. Introduction

Autonomous mobile robot operation in unknown and dynamic environments relies on (1) building a map of the environment based on perceptual data, (2) localizing itself with respect to the map, and (3) autonomous exploration and navigation. Extensive work has been devoted for the past decade to techniques that deal with Simultaneous Localization and Mapping (SLAM), i.e. integrated solutions for the first two problems [1], and also to the action selection problem (e.g. [2]).

For the latter problem, it is important to provide the mobile robot with some kind of alarm that is activated whenever important changes in the environment occur, namely those that may affect its path. Therefore, when the robot revisits some section of the environment, it is worth to compare current perceptual data with previously acquired data, so as to detect significant changes [3]. However, the scope of this problem is not confined to mobile robot navigation. It is certainly important, for instance, for automatic surveillance and security systems [4] or, in general, whenever there is a need to compare two signals of the same type with the aim of detecting novelty. Solving the problem in real-time with huge datasets is quite challenging and requires the development of specific techniques. These techniques aim at achieving two inter-related goals (Fig. 1): first, to detect whether there is some significant change; second, if some significant change exists, to segment the data associated with it.

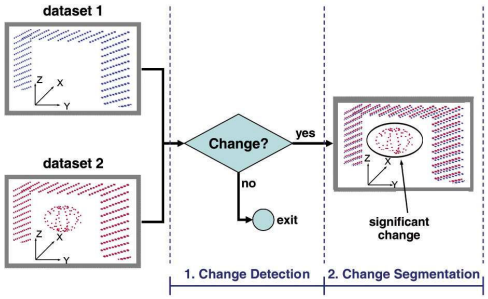

Fig. 1: Change detection and segmentation.

Obtaining a virtual representation of the detected novelty involves the important task of extracting the geometric shape from 3D point clouds. Such virtual representation provides an abstraction of the point data that eliminates much of the redundancy. Besides, primitive shapes can easily be assembled into higher semantic level models that represent dynamic elements of the environment, e.g. persons, boxes or other robots, and improve the later tasks, as tracking or SLAM. In this work, the set of primitive shapes has been limited to three basic ones: sphere, cylinder and plane. Complex models may be obtained by a combination of these primitive shapes [5]. Two different methods are developed in this work. The first one is based on a recent work of Schnabel et al. [5], but applied to the 3D scan data acquired by an autonomous robot. Here, we directly re-sample points using a mixture of Gaussians followed by a RANSAC algorithm to match the set of points to a geometric primitive. In other words, the approach uses samples generated by the GMM in the Euclidean Space and tries to match them to a known shape. The mathematical space of Gaussian mixtures is used in the second approach. Covariance and mean of each Gaussian function associated to a novelty are compared with three geometric primitives according to their ideal covariance and mean. The outputs of this second method are the geometric primitives and rigid transformations that minimize the distance between the two covariance matrices.

The rest of the paper is organized as follows: after briefly showing the state of art of novelty detection and shape retrieval algorithms in Section II, Section III introduces the change detection algorithm. The two proposed algorithms for shape retrieval are presented in Section IV. Experimental results in Section V demonstrate the efficiency and accuracy

This work has been partially supported by the IRPS, EU-FP6-IST-045048 project, by the Spanish Ministerio de Ciencia e innovación (MCINN) Project n. TIN2008-06196 and HP2007-0005, CAPES and FEDER funds.

P. Núñez is member of the ISIS Group, Universidad de Málaga, and is with Dept. Tecnología de los Computadores y las Comunicaciones, Universidad de Extremadura, Spain (e-mail: pmnt@uma.es).

M. Campos and Paulo Drews Jr. are with Dept. Computer Science Federal University of Minas Gerais, Brazil.

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

of the proposed method. Finally, in Section VI, the main conclusions of this approach and future work are drawn.

II. Related work

The proposed approach presents an efficient method for change detection in the robot's environment and the 3-D shape retrieval associated to the change. The behavior of an autonomous mobile robot working in dynamic environments has been intensively researched in the last years. The basic idea behind most of the current navigation systems operating in a dynamic environment deals with removing the dynamic objects in order to improve the navigation and localization tasks [6], [7]. However, these changes in the robot's surrounding may actually be relevant depending of the applications. Thus, Andreasson *et al.* presented a system for autonomous change detection with a security patrol robot [4]. The system uses 3D laser range data and images from a color camera to build a reference model of the environment and then discover changes with respect to that reference model using SIFT features.

Related to the proposed approach, in [3], three different computational techniques for novelty detection were experimentally compared in terms of their novelty discriminating power. Gaussian Mixture Models (GMM) presented the most consistent behavior. Furthermore, a generic metric to quantify changes was formulated by using Earth Mover's Distance (EMD), a distance metric between two data distributions.

The detection of primitive shapes is a common task in many areas of geometry related computer science. In the last decades, a vast number of algorithms has been proposed. Some authors used the well-known Hough's transform [8] to obtain the shape, but their approuch have a high computational cost to compute 3D information. Other techniques are based on region growing [9], which uses a seed region in the scan data which is then grown into neighboring areas. In recent years, some authors have proposed a RANSAC-based shape detection method [10], [5], which is robust for shape retrieval. An excellent review of these methods can be found in [11].

III. Novelty detection Algorithm in 3-D maps

The two main steps of novelty detection depicted in Fig. 1 in general terms are further detailed in Fig. 2 when using both GMM and EMD [3]. First, data in Euclidean space is transformed to the mathematical space of GMM, so as to achieve data compression and efficient comparison using the EMD-based quantification of novelty. Second, if this quantification is above a given threshold, data associated with a novelty is segmented in GMM space and back-propagated to the Euclidean space using the shape retrieval algorithm. The main advantages of using the GMM space are (i) it enables an efficient fusion of point clouds, (ii) its dimensionality is small. A brief description of the novelty detection algorithm is explained in the next subsections.

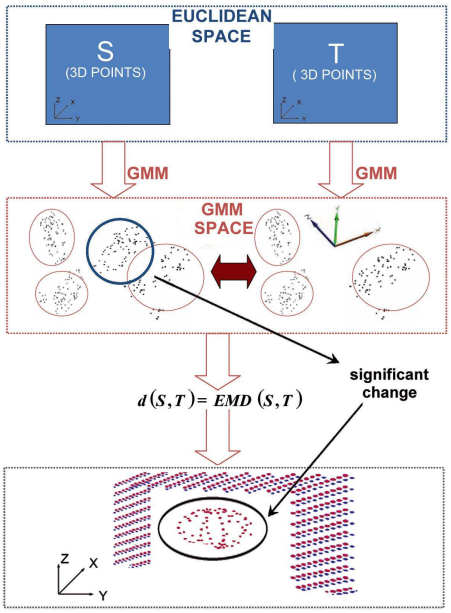

Fig. 2: The two main steps of novelty detection and shape retrieval algorithm proposed in this paper.

A. Mixture of Gaussian functions

A *mixture of Gaussian functions* is a probability density function described by a convex linear combination of Gaussian density functions [12]. Therefore, a function is a mixture of Gaussian functions if it has the form:

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

where the functions g are Gaussian densities which are defined by $\mu_k \in \mathbb{R}^N$ and Σ_k, means and the covariance matrices, respectively, and the coefficients p_k, known as the *mixing probabilities*, which satisfy:

$$p_k \geq 0 \qquad \text{and} \qquad \sum_{k=1}^{K} p_k = 1. \tag{2}$$

Mixtures of Gaussian functions provide good models of clusters of points: each cluster corresponding 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. Conversely, given a set of points in $\mathbb{R}^N$, one can try to find the mixture of Gaussian functions Θ with a certain number of terms that best fits those points, using a method known as *Expectation Maximization* (see section 2.3 in [12]). For the purpose of this paper, Θ denotes the $K(1 + N + N^2)$ dimensional vector containing all the parameters of the given Gaussian mixture:

$$\Theta = ((\theta_1, p_1), \ldots, (\theta_K, p_K)), \tag{3}$$

where

$$\theta_k = (\mu_k, \Sigma_k), \tag{4}$$

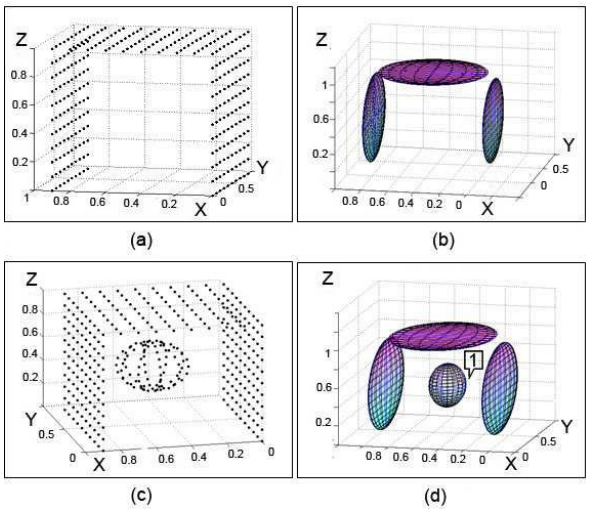

Fig. 3: Novelty detection algorithm: a) ideal 3-dimensional corridor; b) Mixture of Gaussian functions associated to a); c) an object has been moved inside the corridor; and d) Mixture of Gaussian functions associated to c). The novelty detected by the algorithm has been indicated by the label "1".

is a vector containing all the coordinates of the means μ_k and all the entries of the covariance matrix Σ_k. The conditions in Eq. (2) guarantee that f is indeed a density function. Fig. 3a-c illustrate clouds of 3-D points, which describe an ideal corridor where one object has been moved inside. The mixture of Gaussian functions obtained has been shown in Fig. 3b,d. 3-dimensional Gaussians (N = 3) have been detected, each one is associated to the clusters of points (walls, ceiling and the dynamic object).

B. Earth Mover's Distance

The *earth mover's distance* (EMD) [13] can be used to compute the distance between two distributions of points in space for which a distance between points is given. The EMD distance between two sets of points, A and B, is reliably calculated as

$$\text{EMD}(A,B) = \min_{F \in \mathcal{F}(A,B)} \frac{\sum_{i=1}^{m}\sum_{j=1}^{n} f_{ij} d_{ij}}{\min\{W,U\}}, \qquad (5)$$

where

$$A = \{(x_1,w_1),(x_2,w_2),...,(x_m,w_m)\}$$
$$B = \{(y_1,u_1),(y_2,u_2),...,(y_n,u_n)\}$$

, are two sets of n-dimensional weighted points, with $m \leq n$, w and u represent the weight of each point x or y. The total weight W is defined as

$$W = \sum_{i=1}^{m} w_i \quad \text{and} \quad U = \sum_{j=1}^{n} u_j$$

and d_{ij} is the distance from x_i to y_j, and $F = \{f_{ij}\} \in \mathcal{F}(A,B)$, with $\mathcal{F}(A,B)$ being the set of all feasible flows between A and B [13].

Therefore, let $\Theta = ((\theta_1,p_1),\ldots,(\theta_n,p_n))$ and $\Gamma = ((\gamma_1,p_1),\ldots,(\gamma_m,p_m))$ being two mixture of Gaussian functions, associated with two 3-dimensional scan data. In order to measure the presence of changes in the environment, we model each Gaussian function as weighted points $(\theta_i,p_i)_\Theta$ and $(\gamma_j,q_j)_\Gamma$, where θ_i and γ_j mixture of Gaussian functions, and p_i and q_j are the weights associated to each mixture, respectively. Thus, the distance between GMM is calculated as [3]:

$$d_{GMM}(\mathbf{\Theta},\mathbf{\Gamma}) = \text{EMD}\left(\{(\theta_{1\ldots n},p_{1\ldots n})\},\{(\gamma_{1\ldots m},q_{1\ldots m})\}\right). \qquad (6)$$

Algorithm 1 Change segmentation algorithm

1: $d_{GMM} \leftarrow EMDdistance(\Theta,\Gamma)$
2: $\Pi \leftarrow 0$
3: **while** $(d_{GMM} \geq U_{th})$ **do**
4: $\quad x(\Sigma,\mu) \leftarrow SelectGaussianfromGMM(\Theta)$
5: $\quad \Pi \leftarrow \Pi \cup x(\Sigma,\mu)$
6: $\quad \Theta \leftarrow \Theta - x(\Sigma,\mu)$
7: $\quad d_{GMM} \leftarrow EMDdistance(\Theta,\Gamma)$
8: **end while**
9: **return** Π

C. Novelty segmentation to a mixture of Gaussian

Eq. (6) can be used as a quantitative metric to assist the detection of changes in the environment.To achieve this goal, a threshold U_{th} has been defined, which represents the maximum value in order to consider that there is a novelty between two maps [3]. This fixed threshold value has been chosen in the actual implementation of the algorithm, with different real and simulated scenarios. Therefore, the algorithm identifies a novelty in the robot's environment *iff*

$$d_{GMM} \geq U_{th}. \qquad (7)$$

Next, a segmentation process is used to segment these changes in a mixture of Gaussian functions. The overall structure of the method is outlined in pseudo-code in algorithm 1. In each iteration, the algorithm selects a feature $x(\mu,\Sigma)$ from Θ with the greatest quantified change, computed by the *SelectGaussianfromGMM* function. This feature, which is also described by a mixing probability p, is removed from the initial mixture Θ and is also included in the new Gaussian mixture model Π. The two subsets, Θ and Π are recalculated to obtain the new values of the mixing probability. Also, the new mixture of Gaussian is used to obtain a new value of the distance d_{GMM}. The algorithm returns the mixture of Gaussian functions $\Pi \subseteq \Theta$, which identify the changes in the 3-dimensional map.

Fig. 3 illustrates the mixture of Gaussian functions associated with 3-dimensional cluster of points. After applying the proposed algorithm to these two sets of Gaussians, a novelty is detected in the maps (marked as 1 in Fig. 3d).

IV. 3-DIMENSIONAL SHAPE RETRIEVAL

This Section introduces the 3D shape retrieval algorithms used to obtain a model of the detected novelties. The previous

stage obtains a mixture of Gaussian functions, which is related to changes identified in the environment. Let $\Pi = ((\Pi_1, p_1), \ldots, (\Pi_K, p_K))$ be the vector containing the parameters of this Gaussian mixture, which was described in detail in Section III, where $\Pi_k = (\mu_k, \Sigma_k)$ is the means and covariance matrices associated to the novelty cluster of points, and K the number of identified novelties. Next, two methods are presented which achieve the 3-dimensional shape retrieval according to this information.

In this paper, three basic geometric primitives have been used to model novelties, i.e. sphere, cylinder and plane. The first of the methods uses the Euclidean space to generate the shape. Therefore, the mixture of Gaussian functions is used to generate a 3-D points around the identified changes, which are afterwards modeled using a RANSAC algorithm. The second method presents a new strategy to recover a model of the novelty. Thus, the mathematical space of the Gaussian mixture is directly used to achieve this shape retrieval. These two methods will be detailed in the next subsections.

A. Shape retrieval using Euclidean Space

The Gaussian mixture function is a generative model. In other words, it is useful to consider the process of describing a synthetic 3D region using the samples generated from the Gaussian functions. First, one of the samples s is selected at random with prior probability p_k. Next, a data point is generated from the corresponding density Π_k. The corresponding posterior probabilities, $P(k, x)$, can be written using Bayes' theorem, as

$$P(k, x) = \frac{g(\mathbf{x}; \mu_k, \Sigma_k) \cdot p_k}{f(x, \theta)}, \tag{8}$$

where $f(x, \theta)$ is given in Eq. 1. These posterior probabilities satisfy the constraints

$$\sum_{k=1}^{K} P(k \mid x) = 1 \qquad \text{and} \qquad 0 \leq P(k \mid x) \leq 1. \tag{9}$$

By selecting the number of components N to be sampled, it is possible to control the complexity of the synthetic region. In order to improve the shape retrieval algorithm, generated samples are located on the surface of the novelty. Let Σ_k be the covariance matrix associated to the Gaussian. Eigenvectors of this matrix are used to determine the three principal directions of the ellipsoid associated to Σ_k. On the other hand, eigenvalues of this matrix are used to calculate the lengths of these axes from μ_k. Therefore, the resulting samples are generated over the surface of this ellipsoid. Fig. 4b shows the 3-D points clouds generated in the sampling process. In this case, the Gaussian mixture is composed of two terms, which are illustrated as ellipsoids in the Fig. 4a.

Let $S = \{s1, \ldots, s_N\}$ be the set of synthetic 3-dimensional points generated in the previous step. The RANSAC algorithm is applied to this synthetic set to detect the geometric shape of the novelties in the map. The RANSAC paradigm is a well-known strategy to extract shapes from 3D cloud

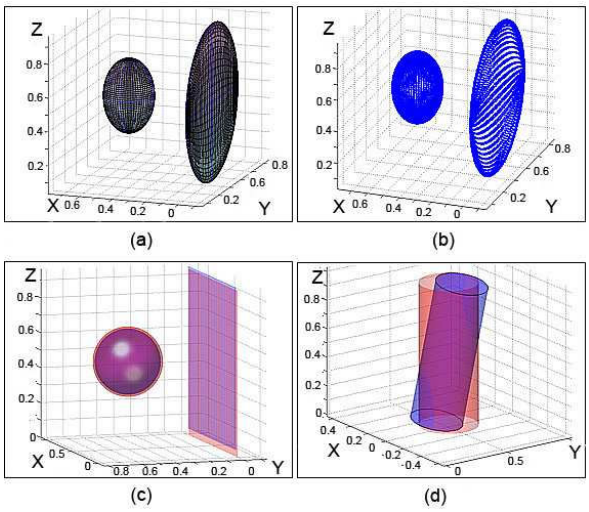

Fig. 4: Shape retrieval algorithms: a) two Gaussian functions which describe the novelty; b) cloud of points generated by the sampling method; c) retrieved shape according to the first method (blue color) and second method (red color) for the example a); and d) retrieved shape for a Gaussian function associated to an ideal cylinder.

of points by randomly drawing minimal sets from the data. The resulting candidates shapes are tested against all the points in the data to determine how many of the points are well approximated by the primitive. After a given number of trials, the shape which approximates the most of the points is extracted and the algorithm continues on the remaining data. Our method used the RANSAC-based algorithm provided by [5], an efficient implementation for recovering the shape of point clouds. The output of the algorithm is a set of primitive shapes $\Psi = \{\Psi_1, \ldots, \Psi_n\}$, with corresponding disjoint set of points $S_{\Psi_1} \subset S$, ..., $S_{\Psi_1} \subset S$ and a set of outliers $R = S \setminus \{S_{\Psi_1}, \ldots S_{\Psi_1}\}$. A detailed description of the algorithm is given in [5]. In this work, the set of possible shapes has been limited to the set $\Psi_{shape} = \{sphere, cylinder, plane\}$. Fig. 4c illustrates the output (in blue color) of the algorithm in the example of Fig. 4a. Fig. 4d shows another example of the representation of a cylinder. In blue color is the shape generated when the covariance Σ_k is associated with an ideal cylindric novelty. As it can be seen in the figure, there is an orientation error in the shape retrieved using Euclidean space, which is due to the loss of information after generating samples from GMM.

B. Shape retrieval based on covariance matrices matching

The shape retrieval algorithm finds the shape that better approximates an ideal basic shape from Ψ_{shape}. To do that, the strategy adopted in this second method uses the mathematical space of the Gaussian mixture model, which is described by the covariance and means of the Gaussian functions. Let Π denote the mixture Gaussians associated with the changes identified in the map. As described in Section III, each novelty k is characterized by a Gaussian

Algorithm 2 Shape retrieval algorithm (Feature space)

1: $\Psi \leftarrow 0$
2: $T \leftarrow 0$
3: $N_{SHAPE} \leftarrow 3$
4: **for** $i = 1$ to N_{SHAPE} **do**
5: **for** $j = 0$ to K **do**
6: $[d^j_\Psi, \Psi^j, T^j] \leftarrow covarianceDistance(\Sigma_j, \Sigma_i)$
7: **end for**
8: $(\Psi_i, T_i) \Leftarrow getBestShape(d_\Psi)$
9: $\Psi \leftarrow \Psi \cup \Psi_i$
10: $T \leftarrow \Psi \cup T_i$
11: **end for**
12: $\mathbf{return}(\Psi, T)$

function, $\Pi_k = (\mu_k, \Sigma_k)$. This paper proposes a new shape retrieval algorithm based on covariance matrices matching. The algorithm is summarized in Algorithm 2. The best model of the shape and the rigid transformation with respect to an ideal shape, T, are the goals of this algorithm. Gaussian functions are matched with each basic shape, by measuring the similarity between their covariance matrices, $d_\Psi = \{d_{sphere}, d_{cylinder}, d_{plane}\}$. The minimum value of d_Ψ determines the shape that best approximates the cloud of points, just as the rigid transformation (line 8 on the algorithm).

Covariance matching is a basic task in measurement design [14]. The goal is to obtain a distance measure of two covariance matrices. The space of covariance matrices is not a vector space and therefore a standard arithmetic difference does not measure the difference between them. But covariance matrices are symmetric and positive semi-definite and then a distance can be formulated based on Riemannian metric. In this paper, the distance metric described by Foerstner and Moonen [14] has been used, which is defined as

$$d(\Sigma_1, \Sigma_2) = \sqrt{\sum_{i=1}^{N} ln^2 \lambda_i(\Sigma_1, \Sigma_2)} \tag{10}$$

where Σ_1 and Σ_2 are the two input covariance matrices, λ represents the generalized eigenvalues of Σ_1 and Σ_2, and N is the dimensionality of the matrices. Here, Σ_1 is the covariance of the Gaussian function which identifies a novelty and Σ_2 is the covariance of a basic shape, i.e. sphere, cylinder or plane. In order to consider possibles rotations and scaling changes of the model, it must be noted that

$$\Sigma_i = T\Sigma_j T^T = (R \cdot L)\Sigma_j (R \cdot L)^T, \tag{11}$$

where T represents the rigid transformation applied to the ideal geometric shape (neither scaling nor rotation), which is composed of rotation and scaling matrices, R and L. In the proposed approach, the translation is directly known with the mean information of each Gaussian.

$$R = \begin{bmatrix} c\psi c\varphi - c\vartheta s\varphi s\psi & c\psi s\varphi + c\vartheta s\varphi s\psi & s\psi s\vartheta \\ -s\psi c\varphi - c\vartheta s\varphi c\psi & -s\psi s\varphi + c\vartheta c\varphi c\psi & c\vartheta s\psi \\ -s\vartheta s\varphi & -s\vartheta c\varphi & c\vartheta \end{bmatrix} \tag{12}$$

$$L = \begin{bmatrix} L_x & 0 & 0 \\ 0 & L_y & 0 \\ 0 & 0 & L_z \end{bmatrix} \tag{13}$$

where ψ, ϑ, φ, are the roll, yaw and pith angles, respectively, c and s stand for the cosine and sine mathematical functions, and L_i is the scaling factor for each axis.

It is possible to minimize Eq. (10) using a least squares minimization method based on Levenberg-Marquardt algorithm, which modifies the rotation and scaling matrices in each iteration. An ininitial guess of the parameters is required to reduce the number of iterations needed to converge and to remove local minima. In order to fulfil these requirements, the algorithm uses a good approximation of the rigid transformation T according to the eigenvectors values of the two covariance matrices. Fig. 4c-d represent the retrieved shape (in red color) using this second method.

V. Experimental Results

The accuracy and computational load of the novelty detection algorithm were measured in our previous work [3]. The authors demonstrated that the GMM-EMD technique is a stable method, in the sense that it presents a lower sensitivity to errors and is able to detect changes with greater reliability.

In this paper, the proposed method has been evaluated using simulated and real data. The algorithms and simulated data were programmed in Matlab, and the benchmark tests were performed on a PC with a 1.6GHz Intel Core2 CPU with 3 Gb of RAM memory and running GNU/Linux. The number of samples generated by the shape retrieval algorithm based on Euclidean space was 3722. The artificial data is formed by a set of 400 points in 3-dimensional space, simulating the readings of a perfect laser range finder in a corridor. A random error, normally distributed with zero mean and variance 0.001, was added to those points. In order to evaluate the algorithm, objects, i.e. cylinders, spheres and planes, are introduced in different poses and with different scale inside the corridor (red color in Fig. 5a-c). A total of 100 different simulated data for each object have been generated for testing.

Real data has been acquired by an Hokuyo URG-04LX laser range finder which is mounted on a Directed Perception PTU-D46 pan-tilt unit. Three different data acquisition areas were used at Instituto de Sistema e Robotica's research laboratory (Figs. 6a-c). The experiments consisted of two captures for each test site. First, a 3-D map was acquired to obtain a first representation of the environment. Afterwards, a novelty was introduced, which is drawn in red color in Figs. 6d-f (an opened door, a person in the corridor and a closed door, respectively). The sizes of those objects were known, and their pose were also calculated using an inertial sensor coupled (XSens MTi) to them and the robot pose.

TABLE I: Comparative study for simulated data.

Algorithm	Euclidean Space	Feature Space
Execution time (s)	0.3491	0.07425
TruePos	0.8330	0.9367
FalsePos	0.1467	0.0400
$\sigma_{\Delta S(S_x,S_y,S_z)}$ [cm]	(12.13, 11.01, 8.09)	(9.11, 7.998, 7.11)
$\sigma_{\Delta R(\psi,\vartheta,\varphi)}$ [deg]	(2.01, 1.12, 3.22)	(1.32, 1.99, 2.88)

TABLE II: Comparative study for real data.

Algorithm	Euclidean Space	Feature Space
Execution time (s)	0.3231	0.3156
TruePos	0.8667	0.9000
FalsePos	0.1912	0.0667
$\sigma_{\Delta S(S_x,S_y,S_z)}$ [cm]	(16.21, 13.24, 10.55)	(12.12, 6.22, 8.19)
$\sigma_{\Delta R(\psi,\vartheta,\varphi)}$ [deg]	(4.21, 3.98, 5.12)	(3.12, 2.01, 3.02)

Finally, in order to obtain statistically significant results, the experiments were repeated ten times for each test site.

The experimental results are focused on the accuracy and processing time of the proposed shape retrieval strategies. Thus, the accuracy measurements for both, simulated and real data, have been defined as

$$\begin{aligned} TruePos &= \frac{NumberTrueShapes}{NumberShapes} \\ FalsePos &= \frac{NumberErrShapes}{NumberShapes}, \end{aligned} \tag{14}$$

where $NumberTrueShapes$ is the number of retrievals of true shapes, $NumberShapes$ is the total number of shapes detected by the algorithm, and $NumberErrShapes$ is the number of retrievals of false shapes. To determine the precision, the shapes obtained by the algorithms have been compared to the real pose and scale of the objects inside the corridor. The results were automatically checked and classified in Tables I and II.

A. Simulated data

Results after applying the algorithms to the simulated data are summarized in Table I. As it can be seen, the average computational load of the two algorithms is different, because the time spent by RANSAC depends on the number of the points and the time to run our method depends on the difference in scale and rotation between the standard shape and the shape to be retrieved. This computation time includes the whole process, that is, change detection, segmentation and shape retrieval. Computational time has been determined using the compiled file in Matlab. Results for the three typical observations in Fig. 5a-c are in Fig. 5d-f, where examples of true and false positives are illustrated for particular cases. The main reasons for failure were: a) error in the *Expectation Maximization* algorithm, due to the proximity of the clusters of points; b) poorly representation of the covariance matrix; and c) ambiguous covariance matrix associated to the clouds of 3-dimensional points. Furthermore, the average error associated with the retrieved shape is larger for the algorithm that uses the re-sampling and RANSAC methods [1].

B. Real data

Table II illustrates the results of the proposed algorithms using real data. Due to the reduced number of experiments, statistical results differs from the simulated data. As it can be observed in the data, the processing time for the algorithm based on the covariance matrices match is larger than using simulated data. In real data, due to the large variation of the shape orientations and the difficulty in getting the initialization to minimize the distance d_Ψ, the computational load is increased. Furthermore, the accuracy of the two proposed approaches decreases for real data. The main reason is the non-ideal shape of the objects with respect to the set of primitives used by the algorithms. Nevertheless, the feasibility of the approach for the purpose of this paper has been demonstrated. Results from three typical observations (Figs. 6a-c) are drawn in Figs. 6j-l (Gaussian functions associated to the novelty have been drawn in Figs. 6g-i).

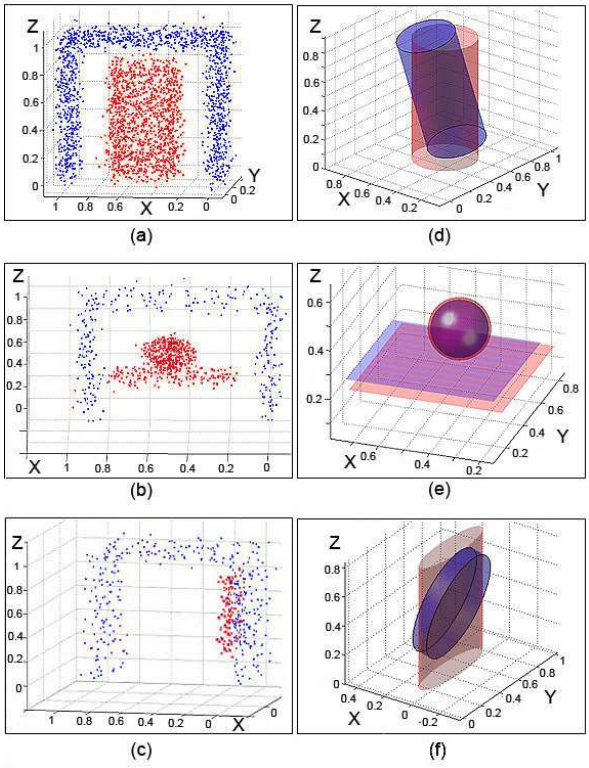

Fig. 5: Simulated data for testing the shape retrieval algorithms: a-c) Simulated observations from an ideal laser of a corridor with an object inside. d-f) Shapes generated by the algorithms (blue: method based on Euclidean space; red: method based on Mathematical space of Gaussian mixture). The results in f) presents a typical problem due to an error in the segmentation process.

VI. Conclusions 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

[1]It is important to note that the worst results for the RANSAC method were obtained when the synthetic object was a cylinder.

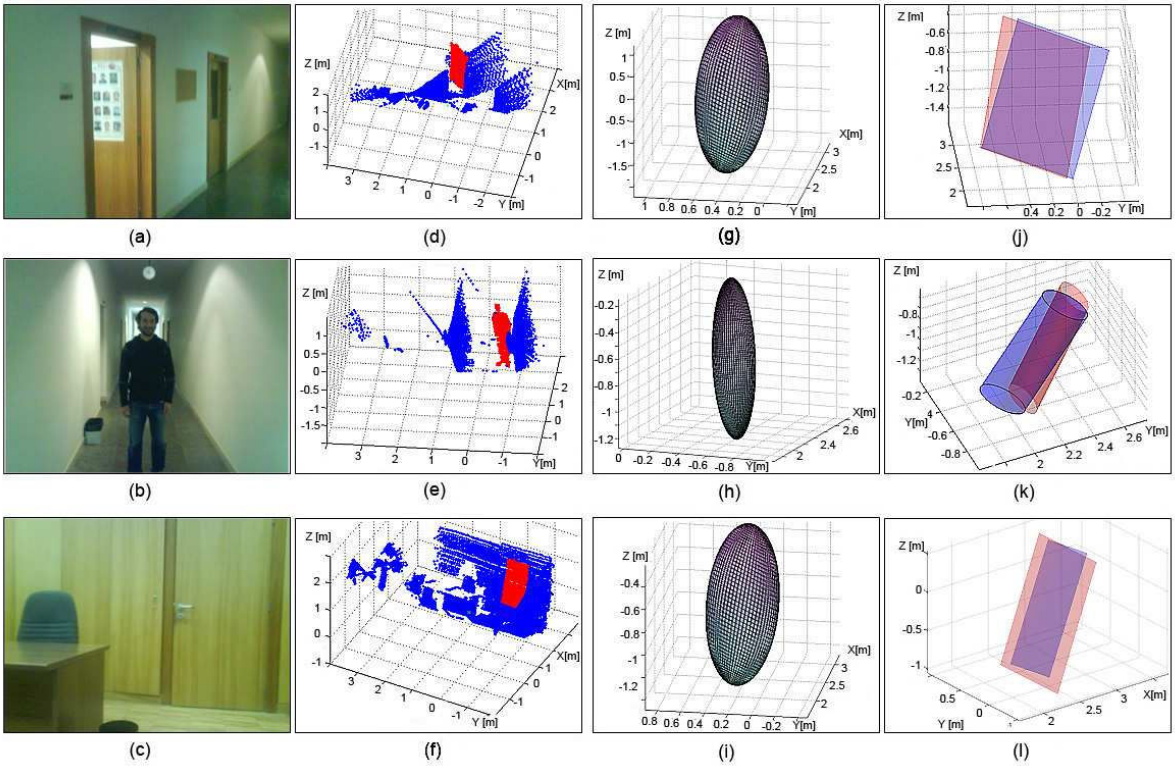

Fig. 6: Real data for testing the shape retrieval algorithms: a-c) Test sites for the real experiments; d-f) Real observations from an Hokuyo laser sensor (3-dimensions); g-i) Gaussian functions associated to the novelty; and j-l) Shapes generated by the algorithms (red: method based on Mathematical space of Gaussian mixture; blue: method based on Euclidean space).

finder and after that, retrieve its shapes using two different methods. Gaussian Mixture Model has been used to obtain a new representation of the point clouds and Earth Mover's Distance is employed to quantify the existence of a novelty in the data. The geometric primitives of the changes has been retrieved using two different methods. The two shape retrieval methods are compared based on the computational cost and accuracy of the retrieved shapes. Experimental results in various real and simulated scenarios demonstrate the feasibility of the approach.

Future work will focus on the extension of the set of geometric primitives (e.g. cones or boxes) or more complex structures (superquadrics) using one or more sensors. The final goal of the work is to obtain a complete system capable of detecting and representing virtual objects in the robot's world which is capable to discriminate various objects.

REFERENCES

[1] S. Thrun, W. Burgard, and D. Fox. "Probabilistic Robotics". *MIT Press*, ISBN 0-262-20162-3, 2005.

[2] R. Rocha, F. Ferreira, and J. Dias. "Multi-robot complete exploration using hill climbing and topological recovery". In *Proc. of 2008 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems* (IROS 2008), pp. 1884–1889, Nice, France, Sep. 2008.

[3] I. Amorim, R. Rocha, and J. Dias. "Mobile Robotic Surveillance Systems: Detecting and Evaluating Changes in 3D Mapped Environments". In *Proc. of 2nd Israeli Conference on Robotics* (ICR2008), Herzlia, Israel, Nov. 19-20, 2008.

[4] H. Andreasson, M. Magnusson, and A. Lilienthal. "Has something changed here? Autonomous Difference Detection for Security Patrol Robots". In *Proc. of IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, pp. 3429–3435, San Diego, USA, 2007.

[5] R. Schnabel, R. Wahl and R. Klein, "Efficient RANSAC for Point-Cloud Shape Detection", in *Computer Graphics Forum* , Volume 26, Number 2, June 2007 , pp. 214-226(13).

[6] D. Fox. "Markov localization for mobile robots in dynamic environments", *Journal of Artificial Intelligence Research*. Vol. 11, pages 391–427, 1999.

[7] P. Núñez, R. Vázquez-Martín, A. Bandera and F. Sandoval. "CF-IDC: A robust robot's self-localization in dynamic environments using curvature information", in *the 14th Mediterranean Electrotechnical Conference* (MELECON08), pp. 330–336, 2008.

[8] D. H. Ballard. "Generalizing the Hough transform to detect arbitrary shapes". *Pattern Recognition*, Vol. 13, n. 2, pp. 111–122, 1981.

[9] M. Vieria and K. Shimada. "Surface mesh segmentation and smooth surface extraction through region growing", *Computer Aided Geometric Design*, Vol. 22, No. 8, pp. 771-792, 2005.

[10] R. Wahl, M. Guthe and R. Klein, "Identifying planes in point-clouds for efficient hybrid rendering", in *the 13th Pacific Conference on Computer Graphics and Applications*, October 2005.

[11] J. Tangelder and R. Veltkamp, "A survey of content based 3D shape retrieval methods", in *Proceedings of the Shape Modeling International 2004*, IEEE computer Society, pp. 145–156, 2004.

[12] P. Pekka and J. K. Kamarainen, J. Ilonen and H. Kälviäinen. "Feature representation and discrimination based on Gaussian mixture model probability densities-Practices and algorithms", *Pattern Recogn.*, Vol. 39, pp. 1346–1358, 2006.

[13] C. Tomasi, Y. Rubner and L. J. Guives. "A metric for distributions with applications to image databases", in *Proc. of the 1998 IEEE International Conference on Computer Vision*, pp. 59–66, 1998.

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