

# A Novel Robust Scene Change Detection Algorithm for Autonomous Robots Using Mixtures of Gaussians

Regular Paper

---

Luis J. Manso<sup>1</sup>, Pedro Núñez<sup>1,\*</sup>, Sidnei da Silva<sup>2</sup> and Paulo Drews-Jr<sup>2</sup>

<sup>1</sup> RoboLab, Universidad de Extremadura, Cáceres, Spain

<sup>2</sup> NAUTEC, Universidade Federal de Rio Grande, FURG, Rio Grande, Brazil

\* Corresponding author E-mail: pnuntru@unex.es

Received 13 Nov 2012; Accepted 04 Nov 2013

DOI: 10.5772/57360

© 2014 The Author(s). Licensee InTech. This is an open access article distributed under the terms of the Creative Commons Attribution License (<http://creativecommons.org/licenses/by/3.0>), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

---

**Abstract** Interest in change detection techniques has considerably increased during recent years in the field of autonomous robotics. This is partly because changes in a robot's working environment are useful for several robotic skills (e.g., spatial cognition, modelling or navigation) and applications (e.g., surveillance or guidance robots). Changes are usually detected by comparing current data provided by the robot's sensors with a previously known map or model of the environment. When the data consists of a large point cloud, dealing with it is a computationally expensive task, mainly due to the amount of points and the redundancy. Using Gaussian Mixture Models (GMM) instead of raw point clouds leads to a more compact feature space that can be used to efficiently process the input data. This allows us to successfully segment the set of 3D points acquired by the sensor and reduce the computational load of the change detection algorithm. However, the segmentation of the environment as a Mixture of Gaussians has some problems that need to be properly addressed. In this paper, a novel change detection algorithm is described in order to improve the robustness and computational cost of previous

approaches. The proposal is based on the classic Expectation Maximization (EM) algorithm, for which different selection criteria are evaluated. As demonstrated in the experimental results section, the proposed change detection algorithm achieves the detection of changes in the robot's working environment faster and more accurately than similar approaches.

**Keywords** Change Detection, Gaussian Mixture Models

---

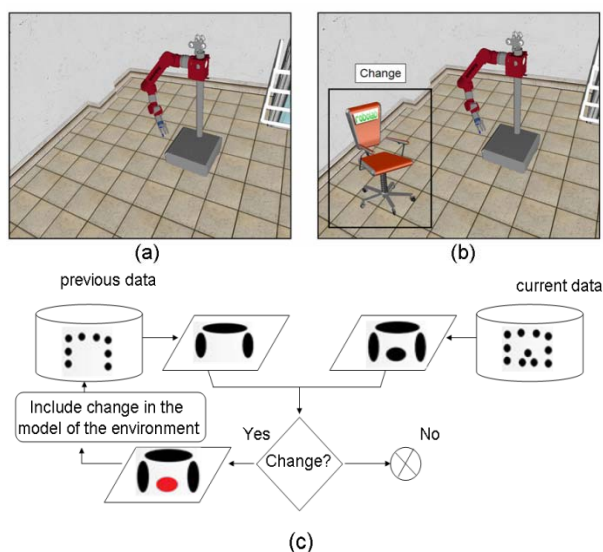
## 1. Introduction

In the last decade, robots and other autonomous systems have been moving away from laboratory setups towards complex real-world environments, which are usually unknown a priori. These environments present dynamic objects (e.g., people or other robots), which are also unknown. Therefore, robots must have capabilities to perceive and generate models of their environments and be aware of the changes in their surroundings. Simultaneous Localization and Mapping (SLAM), one of

the main research focuses in autonomous robot navigation in the last two decades, is starting to exploit this idea. Previous research work shows that models of the environment and/or objects in the scene are an interesting tool for addressing this important task [1] [2].

Change detection is an ability shared by almost all animal species, from the lowest evolutionary level to primates, which is crucial to appropriately adapt behaviour [3]. In the last decade, change detection has emerged as a topic in robotics, where the aim is now to provide mechanisms not only for the representation and storage of environmental information (i.e., short-term or long-term memories), but also for detecting changes in the environment.

Change detection is thus a topic of increasing interest in the autonomous robotics community [4] [5], whose main goal is to provide robots with abilities to autonomously detect and respond suitably to scene changes. For instance, when visiting a known building or walking a known path, the data acquired by the senses (e.g., vision, touch) can provide us with information about possible changes if we can match it to the data we expect. This problem involves different tasks: autonomous acquisition and segmentation of information, strategies for deciding where a relevant change has been introduced in the scene and extraction of semantic models of these changes [6]. Thus, change detection emerges as a mechanism that allows robots to adapt themselves to new situations and to continue their operation, updating their knowledge of the environment and focusing their attention on specific regions of interest. The basic idea behind most current change detection approaches in mobile robotics is to match the obtained data with the expected data available in a map (Figure 1).



**Figure 1.** The main goal of the proposed algorithm is to detect changes in the working environment of the robot (e.g., the chair in b). Problem statement: given the 3D point cloud information acquired by the 3D sensor at time instant  $t$  and a known map on the environment, the robot is required to detect changes in the scene.

In order to improve robustness and to ensure the feasibility of the change detection process, sensor data must be transformed into a more compact form before comparing it with previously acquired data. In this case, the chosen representation heavily determines the performance of the whole task. The model proposed in this paper consists of three major components, sensory, perceptual, and conceptual, which are related with the three sub-problems that arise when detecting changes in robotics:

1. the existence of sensors capable of acquiring a sufficient amount of accurate data;
2. the availability of reliable algorithms capable of extracting high-level representations from this potentially noisy data;
3. the existence of an accurate and robust method to detect changes using this high-level representation.

Considering the first issue in the context of 3D robotic mapping, 3D laser range finders or stereo vision-based systems are commonly used. A 3D laser range scanner is capable of collecting high-quality range data with a very small angular uncertainty. However, they suffer from specular reflections. Applying vision for feature extraction leads to increased CPU usage due to the complexity of the algorithms. In recent years, sensors that combine RGB images with depth information (RGBD sensors) have become widespread in the robotics community. In fact, RGBD cameras allow the acquisition of reasonably accurate mid-resolution depth information at high data rates.

Regarding the second issue, pattern recognition and image processing have inspired different methods for clustering 3D points. Thus, simple methods have been broadly used to support mobile robot operation extracting planar structures, or more compact models [7] [8] [9]. Another possibility is to formulate the 3D clustering problem using a probabilistic approach, e.g., using Mixture Models [10]. Specifically, Mixtures of Gaussian distributions provide good models of point clusters, as demonstrated in previous work by the authors (see [11]).

Finally, with respect to the third question, several metrics have been proposed for detecting changes using the data acquired by the sensors. Typically, the aim is to compare the acquired 3D point cloud with a previously stored one by detecting those points whose minimum distance to the points of the stored point cloud is higher than a threshold. This kind of approach presents several pitfalls (e.g., scale invariance, high computational load, finding the best threshold). In order to reduce the computational cost of this process, more complex metrics, which include statistical information associated with the underlying point distributions, can be used. In [12], the Earth Mover's Distance was proposed as a new metric for solving this case. This metric was employed in a previous

work by the authors, where a greedy algorithm was used to detect changes in the robots' environment [6]. In [11], the authors used structural matching to solve the change detection problem, which was defined as the problem of detecting the maximum clique in a graph. Both approaches are extended and validated in a large set of 3D point clouds in [38].

The main drawback of using Gaussian Mixture Models for segmenting changes in the scene is their strong dependence on the number of Gaussians associated to the map. When this number is known a priori, the complexity of the problem is reduced and the set of Gaussians is estimated using the classic Expectation and Maximization (EM) algorithm [13]. However, when the data is acquired in real time by the robot, there is no information about the number of Gaussians, and thus, it should be estimated, which is usually a hard computational task. In this vein, the paper proposes an algorithm based on the Split-and-Merge paradigm (SM) [14]. The proposal aims to deal with local minimum and the initialization problem of the classic EM. Moreover, it also improves the performance and repeatability of the approach. Therefore, the proposed method focuses on the use of different model selection criteria in order to robustly estimate the number of Gaussians of the mixture, which are evaluated using real and simulated 3D data of indoor environments. This article extends the previous work of the authors of [37], and provides a more thorough experimental evaluation with real datasets.

The rest of the paper is organized as follows. Section 2 summarizes the state of the art in the change detection problem for robotics applications. Section 3 describes the 3D data acquisition and pre-processing stages, as well as the change detection approach and the different criteria used to estimate the number of Gaussians. Experimental results are shown in Section 4. This section also includes a comparison of the proposal with other related approaches. Finally, Section 5 draws the main conclusions and proposes future work.

## 2. Related works

### 2.1 Segmentation of 3D scene data

The problem of modelling arbitrary 3D shapes is common in different areas, such as computer graphics, computer vision or robotics. In computer graphics, the typical approaches deal with triangulation in order to build meshes. These methods allow detailed representations, but the computational cost is high when the surface normals and the neighbourhood are not known a priori. Noise and outliers in the data represent another important challenge. In [16], a method to deal with mesh estimation with noisy data was proposed. After building the mesh, the noise in the data is removed. This method

improves previous approaches; however, it is computationally expensive and heavily dependent on proper mesh estimation.

Another important approach for representing point clouds was proposed in [17]. This allows the surface to be efficiently extracted using implicit surfaces with a low computational cost [18]. However, when the object is not a closed shape, the method is error-prone. The method is based on 3D occupancy grids that restrict the representation to a limited bounding box. Other authors consider basic shapes [19], quadrics [20] and superquadrics [6], but the estimation cost could limit the applicability of these methods to small point clouds.

In this context, clustering using Mixture Models appears an alternative, especially with Gaussian Mixture Models (GMM) [21]. GMM exhibits interesting properties, namely, good compression and description of data, as demonstrated in [22] and [11]. The classical algorithm based on Expectation Maximization was used in [22] for segmenting a 3D point cloud with precise results. However, it was limited by the need for a priori knowledge on the number of Gaussians, and was very dependent on a good initialization.

Some approaches have been proposed to overcome these constraints. In [23], an iterative method was proposed for estimating the number of Gaussians in the model, but it typically suffers from over-fitting. Thus, the present paper proposes an alternative to EM for estimating a Mixture of Gaussians in 3D point clouds without a priori knowledge of the number of models, and the evaluation of different model selection criteria [24].

### 2.2 Scene change detection in robotics

In the last decade, the behaviour of autonomous mobile robots working in dynamic environments has been extensively studied. The common strategy has been to remove dynamic objects (e.g., people, other robots) in order to improve navigation and localization tasks [25]. However, such changes in the robots' surroundings may actually be relevant, depending on the application. In [26], the authors presented a system for automatic change detection with a security patrol robot using 3D laser range data and images from a colour camera, where the texture information was crucial in the change detection stage.

Further important work for detecting changes in surveillance was presented in [27], where the authors used a self-organized network and the concept of habituation to detect changes in sonar data. The work was extended in [28] using visual information, where visual attention was applied through salience maps.

A combination of Gaussian Mixture Models and the Earth Mover's Distance (EMD) algorithm was proposed by [6] to

detect changes in raw 3D point clouds. In spite of the impressive results obtained, the computational load associated to the proposed techniques was not suitable for large datasets. This work was extended in [11], where a structural matching algorithm based on graph theory was used. Both methods are evaluated in [38], where a large set of point clouds is used. Their main limitation is shown to be related to the computational burden. GMMs have also been extensively used to detect novelties in other contexts, but as classifiers [29]. In [4], an interesting survey of previous works on novelty detection using GMMs was presented.

More recently, in [30], the authors used novelty detection to discover new classes in data. Features are selected using the Multiple Discriminant Analysis approach. In this work, an Automatic Guided Vehicle acquires features from a laser scanner and cameras, and then performs the validation of the method. Although this approach has been demonstrated to be efficient, it requires extensive training data for each scenario.

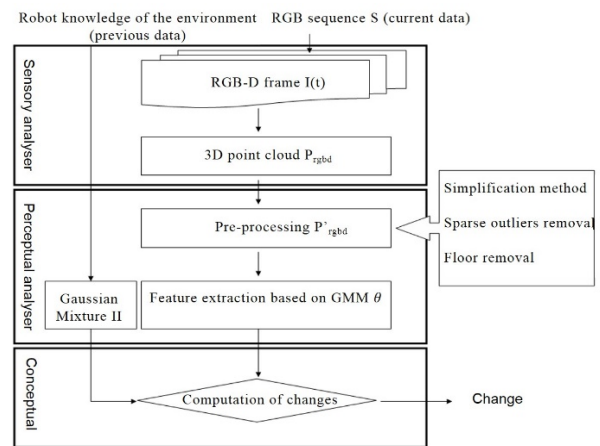
Another important contribution was made by [18], where implicit surfaces were used in order to detect changes in point clouds. Despite the impressive processing time and quality of results, the algorithm is very dependent of the size of the bounding box, and sensitive to complete information about the object and the density parameters, which are manually estimated.

### 3. Scene change detection algorithm

The ability to detect changes in the environment is of prime importance for the survival and functioning of humans and other animals in dynamic and complex environments. In the same way, change detection in robotics is a crucial skill to adapt behaviours accordingly. The proposed system is aimed at detecting significant changes (novelties) and segmenting the data (a set of points) related to them. Figure 2 depicts the flowchart of the change detection approach from an RGBD sequence  $S$ . First, the RGBD sensor acquires the information from the environment. This data is pre-processed in the following stage by two consecutive methods with the aim of reducing the number of points in the 3D map: (i) a downsampling algorithm, and (ii) sparse outliers and ground plane removal methods. For both simplification methods, the Point Cloud Library (PCL) has been used [31].

Computing changes in the Euclidean space by directly processing clouds of points presents several pitfalls. Therefore, it is required to represent 3D data in a more convenient mathematical space for change detection and segmentation. Once the pre-processing stage is completed, the data are converted from a set of 3D points to a more compact representation, Gaussian Mixtures, i.e., clusters extracted using Gaussian Mixture Models. The

use of these clusters enables the description of the environment information in a more convenient way, and the detection and segmentation of novelties. Therefore, novelties are computed and segmented in this mathematical space of GMM.



**Figure 2.** Overview of the change detection algorithm described in this paper

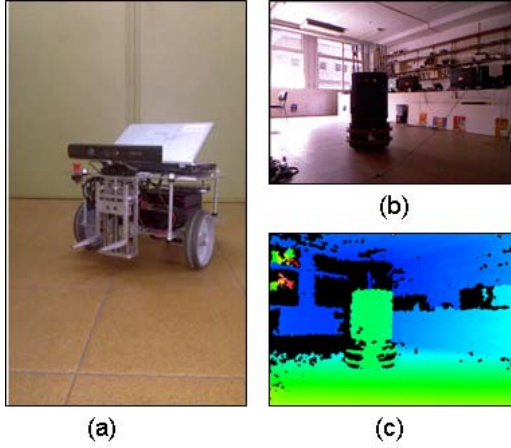
In the next subsections, the complete change detection algorithm is described with details.

#### 3.1 Perception of the robot's surroundings

The input data for the change detection system is a RGBD image acquired using a Primesense sensor mounted on an autonomous mobile robot. The camera provides two different images: a depth image and a colour image. This depth image contains a matrix of pixels, where each pixel represents the distance from the sensed area to the image plane of the sensor. In addition, the colour image is a regular RGB buffer. Both images are acquired in VGA resolution at a rate of 30 fps. Figure 3a illustrates the RGBD sensor mounted on the robot. This sensor provides a video sequence  $S$ , which is composed of RGB and depth images (Figure 3b and Figure 3c, respectively).

The pre-processing stage, which performs point cloud simplification without losing significant information, represents one of the main contributions of this paper. Noise and outliers in the dataset are also reduced. Since the points acquired by the sensor cannot be assumed to be arranged in any particular order, no topological information is available. Thus, the simplification of a large number of points is hard to solve and usually takes a lot of processing time.

The first step of the simplification algorithm is based on the downsampling algorithm Voxel Grid Filter, which reduces the number of points in a way that does not have a big impact on the shape of the surface that the points initially represent [31].



**Figure 3.** a) Kinect sensor mounted on robot RobEx from the RoboLab research group; b) RGB image provided by the sensor; and c) depth image (different RGB values are associated to different distance values)

In order to reduce the noise and outliers in the data, the algorithm Statistical Outlier Removal was used, as proposed in [32]. This uses statistical analysis of the neighbourhood of each point based on a Gaussian probabilistic density function. An outlier is defined using the covariance of each neighbourhood's point. In the presented work, Mahalanobis distance is used for defining the standard deviation in a point as an outlier.

### 3.2 Segmentation of the scene based on Mixture of Gaussians

#### 3.2.1 Gaussian Mixture Models definition

A Gaussian mixture model is a weighted sum of K component Gaussian densities as given by the equation

$$p(\mathbf{x} | \Theta) = \sum_{i=1}^K \omega_i g(\mathbf{x} | \mu_i, \Sigma_i), \quad (1)$$

where  $\mathbf{x}$  is a D-dimensional continuous-valued data vector,  $g(\mathbf{x} | \mu_i, \Sigma_i)$  are the component Gaussian densities, and  $\omega_i$ ,  $i = 1, \dots, K$  are the mixture weights (mixing probabilities) which satisfy the condition:

$$\omega_i > 0 \text{ and } \sum_{i=1}^K \omega_i = 1. \quad (2)$$

Thus, the complete GMM is parameterized by the notation:

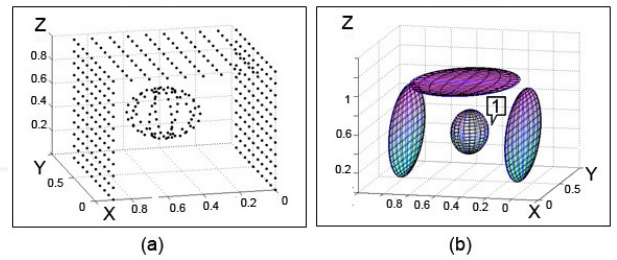
$$\Theta = \{\omega_i, \mu_i, \Sigma_i\}_{i=1, \dots, K}. \quad (3)$$

Given the input data  $\mathbf{S} = \{\mathbf{x}_j\}_{j=1}^N$  composed of N 3D points, a GMM is represented by a collection of Gaussian kernels parameterized by  $\Theta$ , providing a model for each

cluster of points; each cluster is associated to a Gaussian density whose mean is located in the centroid of the cluster and whose covariance matrix estimates the spread of the cluster [10]. Figure 4a describes a synthetically generated ideal hallway, where one object has been placed inside. Figure 4b shows a possible representation based on GMM (a mixture of only 3D Gaussians).

To estimate the parameters of the GMM, a maximization of the following log-likelihood function is performed [10]:

$$L(\Theta | S) = \sum_{j=1}^N \log \sum_{i=1}^K \omega_i g(x_j | \mu_i, \Sigma_i). \quad (\text{eq. 4})$$



**Figure 4.** GMM computation for a synthetically generated hallway: a) cloud of points representing an ideal 3D hallway where a new object was placed inside; GMM associated to the cloud of points a). The Gaussian function representing the novelty is labelled by 1.

#### 3.2.2 Split-EM algorithm

Expectation Maximization (EM) approaches have been commonly used to find the mixture of Gaussian functions that best fits a set of points in  $\mathbb{R}^D$ . The main problem is the computation of the optimal number of Gaussian functions (K). Therefore, the Gaussian mixture modelling becomes a compound problem of the determination of number of Gaussian components and the parameter estimation for the mixture, which is rather challenging. In this paper, a modified version of the work presented in [6] is provided. For this purpose the Split and Merge-EM (SMEM) algorithm has been used [14]. Since the optimal number of Gaussian kernels in each dataset is not known a priori, the approach proposes initializing the mixture model with only one Gaussian, where the mean and covariance matrix are the same as those of the point cloud. This initial Gaussian is dynamically split until the optimal number of Gaussians for the mixture is found. There are two important issues to address here: i) How should a Gaussian be split? ii) When should this process stop? The first question is approached in this paper by using an entropy criterion. In general, the entropy  $H(X)$  of a random variable X with probability mass function  $p(x | \Theta)$  is defined as:

$$H(X) = -E[\log(p(x | \Theta))]. \quad (5)$$

In a similar way, the maximum entropy for a multivariate Gaussian is defined as:

$$H_{\max} = -0.5 \log[(2\pi)^D |\Sigma|] \quad (6)$$

where  $D$  is the dimensionality of the data ( $D = 3$  in the proposed approach) and  $|\Sigma|$  is the determinant of the covariance matrix. It is possible to define the limits for the entropy of each Gaussian component in the mixture using this equation.

The ratio between these two entropies can be used to estimate the density difference for two different times. In each loop of the algorithm, the Gaussian with the smallest ratio is divided into two new Gaussians. The new ratio  $\omega$  is half of the last value. The means of these new two Gaussians are determined by the K-means algorithm [33]. The covariance matrix is estimated using a by-default approach, as in the M-step of the EM algorithm.

In order to solve the second question, i.e., the stopping criterion of the algorithm, three selection model criteria were evaluated. The next sub-section describes in detail the selection criteria used in the proposed approach.

The algorithm proposed can be summarized in three steps:

1. Initially, the GMM is composed of only one Gaussian ( $K = 1$ ). After that, the distribution is updated using the EM algorithm and the log-likelihood is computed using the selection criterion.
2. **Split:** Given the  $K$  and the  $\Theta$  parameters, the Gaussian with the lowest ratio between  $H(X)$  and  $H_{\max}$  is split. The EM updates the parameters  $\Theta$ , where the new GMM has  $K+1$  Gaussians. The log-likelihood is recomputed using the selection criterion.
3. If the new log-likelihood using the selection criterion is larger than the previous one, the GMM is accepted; otherwise, the algorithm returns to step 2.

### 3.2.3 Selection criteria analysis

Selection criteria are used in the literature in order to determine a good representation based on models, minimizing the number of parameters and maximizing the accuracy of the representation. These methods try to find a good trade-off between precision and simplicity in the representation.

In the case of Gaussian Mixture Models, the main goal is to minimize the equation below:

$$\Theta = \operatorname{argmin}_{\Theta} (L_{\text{SC}}(\Theta|S)), \quad (7)$$

where  $L_{\text{SC}}(\Theta|S)$  is the log-likelihood function using selection criterion and  $\Theta$  is the Mixture of  $K$  Gaussians.

The selection criterion defines the  $L_{\text{SC}}(\Theta|S)$  using a penalty function  $P(K)$ :

$$L_{\text{SC}}(\Theta|S) = -L(\Theta|S) + P(K), \quad (8)$$

where  $S$  is the input data, i.e., the point cloud  $S = \{x\}_{i=1}^N$ ; and  $P(K)$  is the penalty function that is increased with the value of  $K$ . This selection criterion tries to avoid over-fitting in the model. The most-used criteria are evaluated in this work to estimate GMM from three-dimensional point clouds. Given the input data, and a mixture of Gaussians with  $J$  free parameters ( $J = 10K-1$ ), next selection criteria are applied inside the Split-EM algorithm to decide when to stop.

- BIC (Bayesian Information Criterion) [34]

$$L_{\text{SC}}(\Theta|S) = -L(\Theta|S) + 0.5J \ln(N). \quad (9)$$

- MDL (Minimum Description Length) [23]

$$L_{\text{SC}}(\Theta|S) = -L(\Theta|S) + \frac{D}{2} \sum_{i=1}^K \log\left(\frac{N\omega_i}{12}\right) + \frac{K}{2} \log\left(\frac{N}{2}\right) + \frac{K(D+1)}{2}. \quad (10)$$

- AIC (Akaike Information Criterion) [35]:

$$L_{\text{SC}}(\Theta|S) = -L(\Theta|S) + 2J. \quad (11)$$

### 3.3 Change detection and segmentation

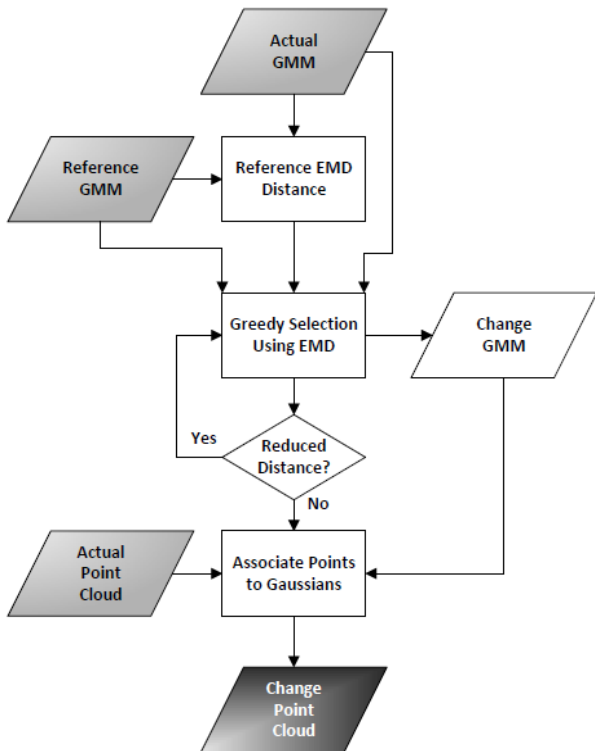
Once both datasets are transformed into Gaussian Mixtures, the aim is to process them in order to estimate the changes in the scene, i.e., detecting and segmenting novelties in the robot's surrounding. The Earth Mover's Distance (EMD) was proposed in [12] as a metric for measuring distances between two distributions of points in the space for which a distance between points is given. In the mathematical space of Gaussian Mixture:

$$d_{\text{gmm}}(\Theta, \Phi) = \text{EMD}(\{\omega_i, \mu_i, \Sigma_i\}_{i=1, \dots, K}, \{\omega_j, \mu_j, \Sigma_j\}_{j=1, \dots, K'}) \quad (11)$$

where  $\Theta$  and  $\Phi$  are two mixture of Gaussians with  $K$  and  $K'$  Gaussians, respectively. If the distance  $d_{\text{gmm}}$  is higher than a threshold, the algorithm detects the change in the scene. In this paper, a greedy-EMD algorithm has been applied in order to avoid the use of a fixed threshold [6].

The overall structure of the EMD-based algorithm used in this paper is shown in Figure 5. The method not only

detects changes, but segments them. The set of points associated to the change is retrieved using the posterior probability. Initially, the algorithm computes the EMD distance between the GMMs that represents the reference map and the actual map. In each iteration, the algorithm selects the Gaussian from the actual GMM with the greatest quantified change. Furthermore, it generates a new GMM called Change GMM. The best Gaussian is removed from the mixture and is included in the Change GMM. The EMD distance between the reference GMM and the Actual GMM without the Gaussians of the Change GMM is computed. If this distance is smaller than the reference distance, the reference distance is updated and the process restarts, as shown in Figure 5. The Change GMM is then defined as the Gaussians that represent the changes. The points in the actual point cloud are segmented using the Change GMM and the posterior probability. If the Change GMM is empty, the algorithm assumes that no novelties exist, implying that the two GMMs are similar. The posterior probability allows the system to identify the topological relation between the segmented regions. This kind of information could be useful both for recognition and for identification, providing a means to build a semantic representation of the environment. See [6] for more details about this algorithm.



**Figure 5.** Change detection and segmentation algorithm based on a greedy-EMD method

#### 4. Experimental results

The proposed method was evaluated using real and simulated data. The algorithms were developed in C++ and

the benchmark tests were performed on an Intel Core2Duo 2.4GHz CPU with 4GB of RAM running GNU/Linux.

Real data were acquired using a RobEx robot (Figure 3a). The robot RobEx is a differential robot designed by RoboLab at the University of Extremadura. For the experiments described in this paper, a Kinect was mounted on top of it and set up for acquiring RGBD images at 30 fps. Figures 6a-b show the first and second test areas within the RoboLab facilities. Both indoor test areas provided a real environment for the evaluation of the algorithm.

The simulated data were acquired using RCIS (RoboComp InnerModel Simulator) [36]. RCIS is a 3D robot simulator designed for use in academia, in early stages of development, and, mainly, for research purposes. One of its most remarkable features is that it enables users to control the noise produced by the simulated sensors and actuators. This ability can be used to test how robust algorithms are against noise and, if noise is set to zero, to differentiate between problems dealing with noise and algorithm errors. RCIS implements all the interfaces of the hardware abstraction layer of RoboComp (e.g., Camera, Laser, RGBD, Joint Motor Bus), which represent most of the common hardware used in autonomous robotics. For the tests described in this paper three RGBD sensors with different noise levels were simulated within an indoor scenario. The use of this simulator allows an evaluation of the algorithms according to different sensor noise values.



(a)

(b)



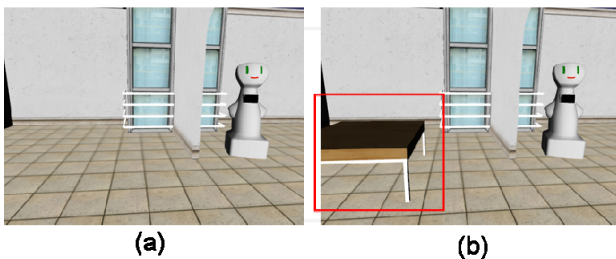
(c)

**Figure 6.** a) First test area, an indoor scenario in which the robot Nomad has been placed to represent a change; b) a pair of chairs in the room, the second of the experiments; and c) the simulated indoor scenario used for the first experiment using the RCIS simulator

The experiments were focused on the evaluation of the algorithm in terms of robustness and computational resources. A comparative study of the different selection criteria was obtained as a result.

#### 4.1 Simulated scenario

As aforementioned, the RCIS RoboComp Simulator was used in order to evaluate the robustness of the algorithm regarding different sensor noise values. Figure 7 illustrates the environment used for this experiment. The image was acquired in the same robot pose; however, a change has been introduced in the scene (i.e., a table, as shown in Figure 7b).



**Figure 7.** a) An indoor environment simulated by RCIS RoboComp Simulator; b) the same scenario, but where a new element has been introduced

Three different RGBD sensor noise values were used in the experiments. The distance noise was simulated using a normal distribution with  $\mu = 0$ , and variance values as defined in Table 1.

Test	Sigma (m)
0	0.001
1	0.01
2	0.1

**Table 1.** Depth noise values used for the experiment in the simulated test area

Table 2 illustrates the number of points associated to the scene acquired by the simulator for each experiment after the simplification stage. In Table 4,  $n_1$  represents the number of points after the simplification stage for the scene presented in Figure 7a. Similarly,  $n_2$  describes the number of points after the simplification stage for the scene presented in Figure 7b.

Test	$n_1$	$n_2$
0	14586	14033
1	19167	18271
2	9528	8823

**Table 2.** Number of points obtained after the simplification stage for the experiments described in this paper

The same experiments were achieved for each selection criterion AIC, BIC and MDL. The results are summarized in Table 3, where  $K_i$  represents the number of Gaussians in the

mixture, for the scene presented in Figure 7a and Figure 7b, respectively. The computational load of the change detection algorithm is also shown in Table 3. The number of changes detected in the scene is presented in Table 4. The ground truth of the change is manually selected in the scene. The change corresponds to only one Gaussian in all the tests.

As shown in Tables 3 and 4, the best selection criterion is the Minimum Description Length. MDL allows the algorithm to segment the 3D scene into a representative Mixture of Gaussians with low computational cost. The GMM obtained using the MDL criterion is enough for detecting the change in the scene.

Criterion	Test 0		Test 1		Test 2	
	$K_1$ / $K_2$	Time (s)	$K_1$ / $K_2$	Time (s)	$K_1$ / $K_2$	Time (s)
AIC	13/ 14	10.969	16/ 18	14.14	20/ 20	7.125
BIC	6/ 6	3.472	6/ 8	5.22	16/ 18	6.286
MDL	4/ 4	<b>2.813</b>	6/ 7	<b>1.93</b>	7/ 7	<b>0.924</b>

**Table 3.** Comparative study of the different selection criteria used in the segmentation algorithm. Number of Gaussians in the mixture and computational time are shown

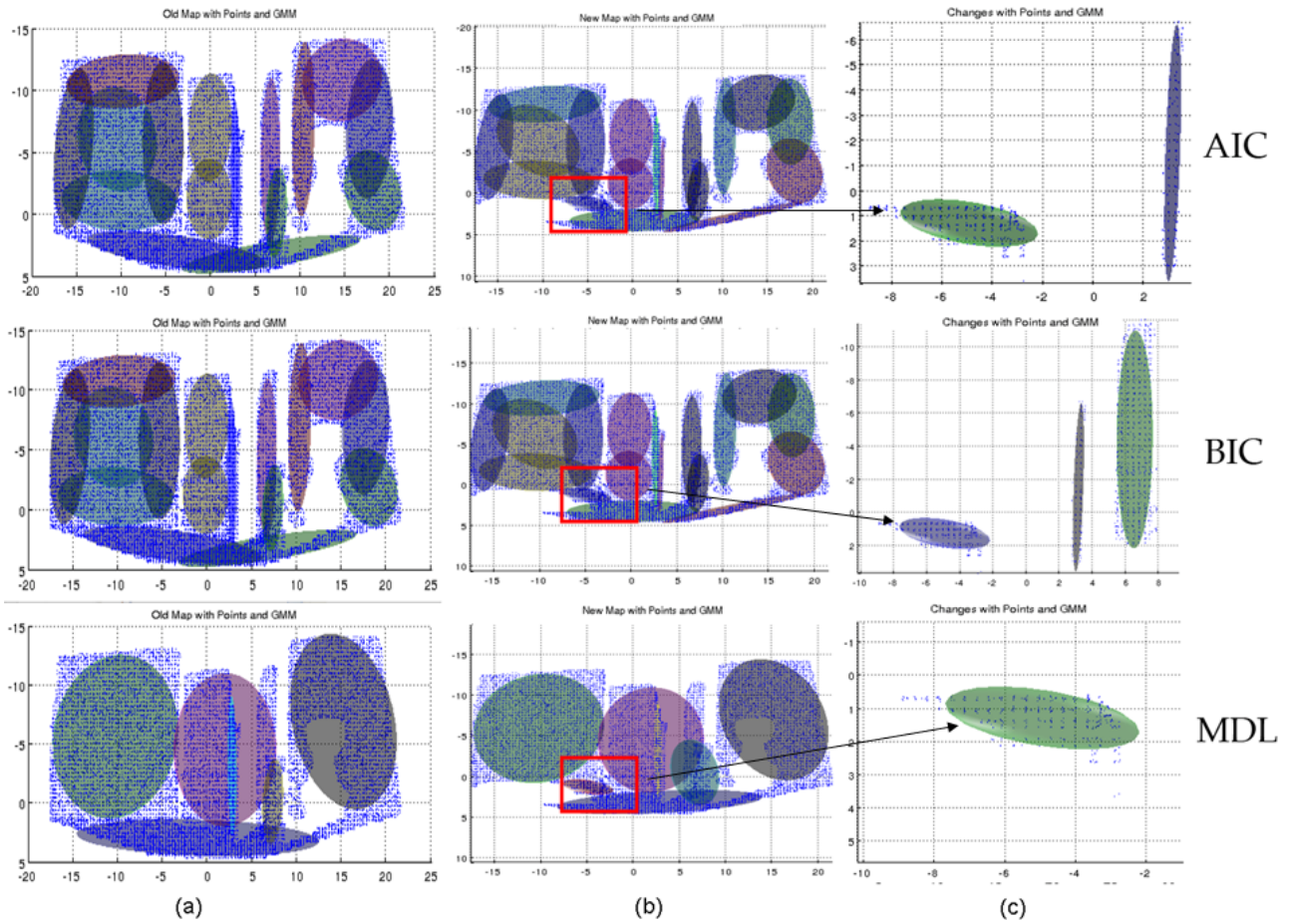
Criterion	Test 0	Test 1	Test 2
AIC	1	2	0
BIC	1	3	4
MDL	1	1	1

**Table 4.** Number of Gaussians detected as changes in the scene

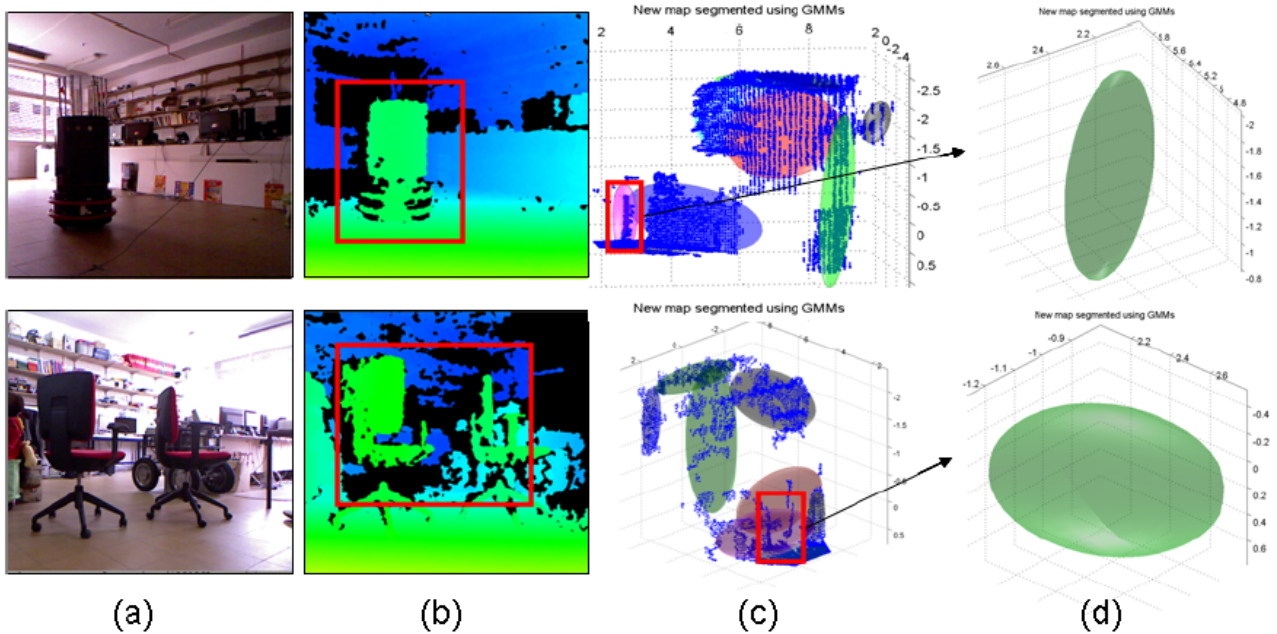
Figure 8 shows the results after applying the change detection algorithm in the simulated environment for each selection criterion and for a Gaussian noise with  $\mu = 0$ , and variance  $\sigma = 0.01$  m (similar results were obtained with other noise values). Figure 8a illustrates the Mixture of Gaussians associated to the scenario shown in Figure 7a. Similarly, Figure 8b draws the GMM of the scenario described in Figure 7b. Finally, Figure 8c illustrates the detected change, i.e. the Gaussian of the mixture after evaluating the greedy-EMD algorithm. Only the MDL selection criterion correctly segments the change. The coloured ellipsoids shown Figures 8b and 8c are generated using 1.5 standard deviation based on the covariance matrix, and the point clouds are shown by blue dots.

Table 5 demonstrates that the MDL criterion is the most adequate for the segmentation algorithm. The MDL criterion allows faster segmenting of the point cloud, using a smaller number of Gaussians in the mixture. As shown in Figure 9, MDL also obtains an accurate segmentation of the change in the scene.





**Figure 8.** a) Map of the environment and segmentation based on Gaussian Mixture Models; b) map of the environment after placing a table (change) in the environment; and c) segmentation of the change in the scene. Only the MDL selection criterion correctly segments the change.



**Figure 9.** a) First test area, an indoor scenario in which the robot Nomada is placed to represent a change; b) a depth image where a pair of chairs in the room; and c) a simulated indoor scenario using an RCIS RoboComp 3D simulator. The number of points in both tests are  $n_1 = (19798, 13140)$  and  $n_2 = (18437, 10507)$ , respectively.

Criterion	Test 0			Test 1		
	K <sub>1</sub> / K <sub>2</sub>	time(s)	it	K <sub>1</sub> / K <sub>2</sub>	time(s)	it
AIC	12 20	18.96	570	12 16	9.18	643
BIC	12 20	18.10	563	12 16	8.9	620
MDL	7 8	5.64	309	6 8	3.76	368

**Table 5.** Comparative study of the different selection criteria used in the segmentation algorithm using real point clouds. The number of Gaussians in the mixture and computational time, as well as the number of iterations, are shown.

## 5. Conclusions and future work

In this paper, a change detection and segmentation algorithm based on Mixture of Gaussians has been described for use in autonomous robots. It has been tested using both real and simulated RGBD data. Real data are acquired using the Kinect sensor, which provides RGB and depth (D) images. The 3D point clouds are transformed into a new high-level representation based on a Gaussian Mixture Model. Computing changes in this new Mathematical Space are faster and have some advantages with respect to using the Euclidean space. Robustness and the improvement of the computational load of the approach have been achieved thanks to an efficient estimate of the number of Gaussians of the mixture. Therefore, different selection criteria have been evaluated in order to make the segmentation process in the Split and Merge method more robust. Experimental results obtained with the application of the proposed algorithm with both real and simulated data demonstrate the reliability of the method in real robot environments.

Future work will focus on including an attentional model with a novelty filter in order to process only salient locations, rather than the whole scene. Visual information can also be included in the change detection process in order to improve the estimation in structured environments. For instance, it is interesting to take into account floor and roof removals, or other main planes (e.g., walls). This helps to improve the performance of such systems by allowing them to allocate extra processing resources to those stimuli deemed to be of higher potential relevance. Applications to underwater side sonar scanning data will also be evaluated.

## 6. Acknowledgements

This work has been partially supported by the MICINN Project TIN2012-38079-C03-01 and by the Extremadura Government. This work has also been partially supported by CNPQ, FAPERGS, CAPES, and PDE-

FURG. It is a contribution of the Brazilian National Institute of Science and Technology - INCT-Mar COI funded by CNPq grant number 610012/2011-8.

## 7. References

- [1] Ranganathan A. and Dellaert F. Semantic modeling of places using objects. In *Robotics: Science and Systems 3*, 27-30, 2007.
- [2] Civera J., Gálvez-López D., Riazuelo L., Tardós J.D. and Montiel J.M. Towards Semantic SLAM using a Monocular Camera. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 1277-1284, 2011.
- [3] Grandjean D. and Peters C. Novelty processing and emotion: conceptual developments, empirical findings and virtual environments. *Emotions-Oriented Systems. The Humaine Handbook*. Springer Berlin Heidelberg, 441-458, 1998.
- [4] Markou M. and Singh S. Novelty detection: A review. Part 1: Statistical approaches. In *Signal Processing* 83, 2481-2497, 2003.
- [5] Chandola V., Banerjee A. and Kumar V. Anomaly detection: A survey. In *ACM Computing Surveys*, (CSUR) 41, 1-58, 2009.
- [6] Drews-Jr P., Nuñez P., Rocha R., Montenegro M.F. and Dias J. Novelty Detection and 3D Shape Retrieval using Superquadrics and Multi-Scale Sampling for Autonomous Mobile Robots. In *International Conference on Robotic and Automation, ICRA*, pp. 3635-3640, 2010.
- [7] Loménie N. A generic methodology for partitioning unorganised 3d point clouds for robotic vision. In *Proceedings of the Canadian Conference Computer and Robot Vision*, pp. 64-71, 2004.
- [8] Mufti F., Mahony R. and Heinzmann J. Robust estimation of planar surfaces using spatio-temporal ransac for applications in autonomous vehicle navigation. In *Robotics and Autonomous Systems* 60, 16-28, 2012.
- [9] Schlemmer M., Biegelbauer G. and Vincze M. Rethinking Robot Vision - Combining Shape and Appearance. *International Journal of Advanced Robotic Systems (IJARS)* 4(3), 256-270, 2007.
- [10] Bishop C. *Pattern Recognition and Machine Learning*, Springer, 740p, 2009.
- [11] Nuñez P., Drews-Jr P., Bandera A., Rocha R., Campos M. and Dias J. Change detection in 3D environments based on Gaussian Mixture Model and robust structural matching for autonomous robotic applications. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 2633-2638, 2010.
- [12] Tomasi C., Rubner Y. and Guives L. A metric for distributions with applications to image databases, In *Proceedings of International Conference on Computer Vision (ICCV)*, pp. 59-66, 1998.

- [13] Dempster A.P., Laird N.M. and Rubin D.B. Maximum Likelihood from Incomplete Data via the EM Algorithm. *Journal of the Royal Statistical Society, Series B (Methodological)* 39(1), 1–38, 1977.
- [14] Li Y. and Li L. A novel split and merge EM algorithm for Gaussian Mixture Model, *Natural Computation*. In *Fifth International Conference on Natural Computation ICNC '09* 6, 479–483, 2009.
- [16] Vieira A.W., Alves Neto A., Macharet D. and Campos M. Mesh denoising using quadric error metric. *XXIII Brazilian Symposium on Computer Graphics and Image Processing (SIBGRAPI'10)*, Gramado, RS, Brazil, 247–254, 2010.
- [17] Bloomenthal J. *Introduction to Implicit Surfaces*, Morgan Kaufmann Publishers Inc, San Francisco, CA, USA, 1997.
- [18] Vieira A.W., Drews-Jr. P. and Campos M. Efficient change detection in 3d environment for autonomous surveillance robots based on implicit volume, *The 2012 IEEE International Conference on Robotics and Automation, (ICRA'12)*, 2999–3004, 2012.
- [19] Schnabel R., Wahl R. and Klein R. Efficient Ransac for point-cloud shape detection. *Computer Graphics Forum* 26(2), 214–226.
- [20] Pauling F., Bosse M. and Zlot R. Automatic segmentation of 3d laser point clouds by ellipsoidal region growing. In *Scheding S. (ed.), Australasian Conference on Robotics and Automation (ACRA)*, 10p, 2009.
- [21] McLachlan G.J. and Peel D. *Mixture Models: Inference and Applications to Clustering*, Wiley-Interscience, New York, 2000.
- [22] Núñez P., Drews-Jr P., Rocha R., Campos M., Dias J. Novelty detection and 3D shape retrieval based on Gaussian Mixture Models for autonomous surveillance robotics. In *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 4724–4730, 2009.
- [23] Figueiredo M. and Jain A. Unsupervised learning of finite mixture models. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 23, 381–396, 2002.
- [24] Hu X. and Xu L. Investigation on several model selection criteria for determining the number of clusters. *Neural Information Processing - Letters & Reviews* 4(1), 1–10, 2004.
- [25] Thrun S., Fox D., Burgard W., Dellaert F. Robust Monte Carlo localization for mobile robots. *Artificial Intelligence* 128, 99–141, 2001
- [26] Andreasson H., Magnusson M., Lilienthal A.J. Has something changed here? Autonomous difference detection for security patrol robots. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 3429–3435, 2007.
- [27] Marsland S., Nehmzow U., Shapiro J. Environment-specific novelty detection. In *Proceedings of the Seventh International Conference on Simulation of Adaptive Behavior on From Animals to Animats (ICSAB)*, pp. 36–45, 2002.
- [28] Vieira Neto H., Nehmzow U. Visual novelty detection with automatic scale selection. *Robotic and Autonomous Systems* 55, 693–701, 2007.
- [29] Tarassenko L., Hayton P., Cerneaz N., Brady M. Novelty detection for the identification of masses in mammograms. In *Proceedings of International Conference on Artificial Neural Networks (ICANN)*, pp. 442–447, 1995.
- [30] Sofman B., Neuman B., Stentz A., Bagnell J.A. Anytime online novelty and change detection for mobile robots. *Journal of Field Robotics* 28, 589–618, 2011.
- [31] Rusu R.B. and Cousins S. 3D is here: Point Cloud Library (PCL). *IEEE International Conference on Robotics and Automation (ICRA)*, Shanghai, China, 1–4, 2011.
- [32] Rusu R., Marton Z., Blodow N., Dolha M. and Beetz M. Towards 3D point cloud based object maps for household environments. *Robotics and Autonomous Systems* 56(11), 927–941, 2008.
- [33] MacQueen J.B. *Some Methods for Classification and Analysis of Multivariate Observations*. *Proceedings of 5th Berkeley Symposium on Mathematical Statistics and Probability*. University of California Press, Berkeley, CA, USA, pp. 281–297, 1967.
- [34] Fraley C. and Raftery A.E. How Many Clusters? Which Clustering Method? Answers Via Model-Based Cluster Analysis. *The Computer Journal* 41(8), 578–588, 1998.
- [35] Akaike H. A new look at the statistical model identification. *IEEE Transactions on Automatic Control* 19(6), 716–723.
- [36] Manso L., Bachiller P., Bustos P., Núñez P., Cintas R. and Calderita L. RoboComp: a Tool-based Robotics Framework. In *Proceedings, SIMPAR Second International Conference on Simulation, Modeling and Programming for Autonomous Robots*, pp. 251–262, November 2010, Darmstadt, Germany.
- [37] Drews-Jr P., Rocha R., Campos M., Dias J. Fast and Adaptive 3D Change Detection Algorithm for Autonomous Robots based on Gaussian Mixture Models, In *International Conference on Robotic and Automation, ICRA*, pp. 4670–4675, 2013.
- [38] Drews-Jr P., Núñez P. Rocha R., Montenegro M.F. and Dias J. Novelty detection and segmentation based on Gaussian mixture models: A case study in 3D robotic laser mapping. *Robotics and Autonomous Systems*, available online, July 2013.