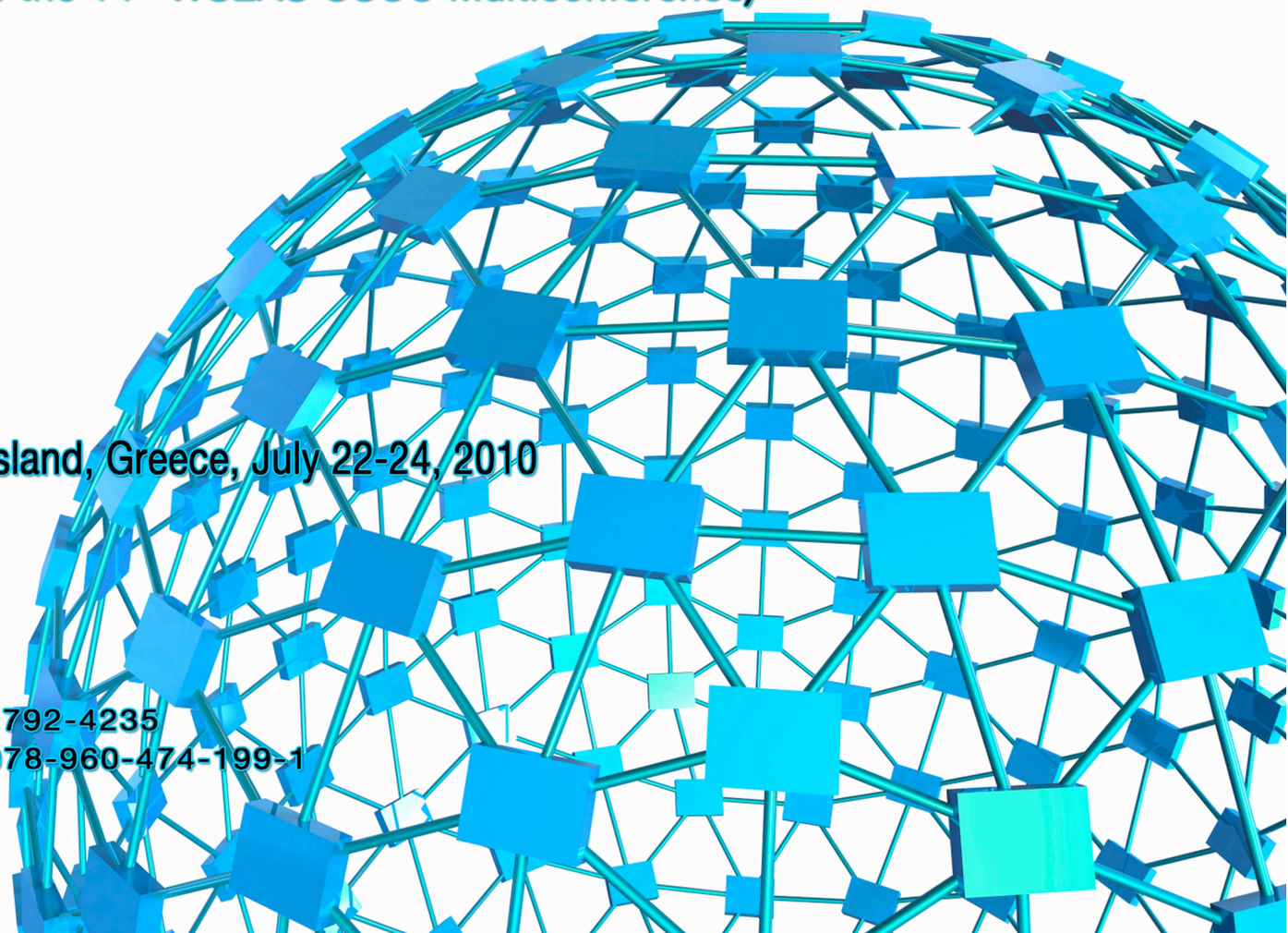
Editors: N. Mastorakis, V. Mladenov, Z. Bojkovic
Associate Editor: V. Vasek

Latest Trends on Systems

VOLUME I

14th WSEAS International Conference on Systems
(Part of the 14th WSEAS CSCC Multiconference)



Corfu Island, Greece, July 22-24, 2010

ISSN: 1792-4235
ISBN: 978-960-474-199-1

Latest Trends on Systems

LATEST TRENDS on SYSTEMS (Volume I)

14th WSEAS International Conference on SYSTEMS
(Part of the 14th WSEAS CSCC Multiconference)
(Volume I)

Corfu Island, Greece
July 22-24, 2010

Mathematics and Computers in Science Engineering
A Series of Reference Books and Textbooks

Published by WSEAS Press
www.wseas.org

ISSN: 1792-4235
ISBN: 978-960-474-199-1

LATEST TRENDS on SYSTEMS (Volume I)

14th WSEAS International Conference on SYSTEMS (Part of the 14th WSEAS CSCC Multiconference) (Volume I)

Corfu Island, Greece, July 22-24, 2010

Mathematics and Computers in Science Engineering
A Series of Reference Books and Textbooks

Published by WSEAS Press
www.wseas.org

All papers of the present volume were peer reviewed by two independent reviewers. Acceptance was granted when both reviewers' recommendations were positive.
See also: http://www.worldses.org/review/index.html

ISSN: 1792-4235
ISBN: 978-960-474-199-1

World Scientific and Engineering Academy and Society

Table of Contents

Map Fusion in an Independent Multi-robot Approach

MONICA BALLESTA, ARTURO GIL, ÓSCAR REINOSO, LUIS PAYÁ and LUIS M. JIMÉNEZ
Miguel Hernández University
Department of Industrial Systems Engineering
Avda. de la universidad s/n, 03202, Elche
SPAIN
$m.ballesta||arturo.gil||o.reinoso||lpaya||luis.jimenez@umh.es$

Abstract: This paper concentrates on the study on the map fusion problem in the context of a multi-robot map building approach. Concretely it is seen as one of the steps towards the independent multi-robot map building. In the situation proposed a set of several robots performs map building tasks without the notion of other robots' existence. Each robot builds its own local map using its observations and estimates its path independently. As a result, there will be a set of local maps that can be fused into a global one. This is the case when the map fusion takes importance. Particularly, we focus our experiments on landmark-based maps constructed using visual information and by means of a particle filter. When fusing two maps, we consider the uncertainty of the landmarks integrated by each different robot to its map.

Key–Words: SLAM, map fusion, visual landmarks, particle filter, map alignment

1 Introduction

Map building is one of the fundamental tasks that has to be accomplished by a robot to be considered as autonomous. The capability of building a map of the environment while simultaneously the robot localizes in it is known as SLAM (Simultaneous Localization and Mapping) and has received great attention over the last years [19].

A single robot is able to carry out the construction of a map. However, this task will be more efficiently performed if there is a team of robots that cooperate in the consecution of this objective [15, 18]. In this case the map building will be performed more quickly and robustly than with a single robot [22]. However, the trajectories of several robots have to be estimated meanwhile information from different entities is fused to estimate a single map. As a consequence, the dimensionality of the problem is higher.

Regarding the sensors used to extract information from the environment, some authors employ range sensors such as LASER [31, 32] or SONAR [34]. However, there is an increasing interest on using cameras as sensors [30]. This approach is denoted as visual SLAM [33, 7]. These devices obtain a higher amount of information from the environment and are less expensive than other sensors such as LASER. Moreover, 3D information can be directly obtained when using stereo vision [10]. Finally, in order to build the maps, a recent proposal is the FastSLAM algorithm [25]. The main idea of this algorithm is the use of a particle set that represents the uncertainty in the pose of the robot. Each particle is an hypothesis of the real path followed by the robot and has an associated map of the environment. The SLAM problem is seen as the sum of two fundamental aspects: the estimate of the pose of the robot and the estimate of the map. Although these aspects are intrinsically related, they can be considered separately. That is to say, if the robot's path is known, then the estimate of the map would be trivial. In a similar way, if the map is known, it would be easy to localize the robot in it. The FastSLAM algorithm divides the SLAM problem into a localization problem and several individual estimates of the map. These steps are repeated successively during the SLAM process.

Our work focuses towards the approach in which there is a team of robots that colaborates in the construction of a map of the environment. In this approach the map and the trajectories are not built jointly such as in other multi-robot proposals [9]. On the contrary, we propose an alternative solution in which the robots initiate the SLAM process independently, i.e., they have no knowledge about other robots' poses and observations. The map building can be performed without knowing the relative positions of the robots. The SLAM problem is therefore solved by means of several independent particle filters. After a while, each robot will have built a local map with its own reference system. In order to obtain a global map, the set of local maps have to be fused into a single one. In this paper, we focus on this step. First, it is neces-

sary to estimate the relative position in order to find a common reference system for the local maps. This is denoted as map alignment and consist in computing the tranformation that relates two reference systems. This is done by establishing correspondences between the local maps. Finally the global map is obtained in the map merging step, in which the data is fused. The study of the map alignment and map merging, tackled in this paper, is a necessary preliminary step in order to achieve an independent multi-robot SLAM platform.

2 Related work

Different solutions to the multi-robot SLAM problem have emerged so far. These solutions can be classified into two different groups:

1. Solutions in which the estimate of the map and the robot trajectories is performed jointly. In this case, the construction of a single map is centralized using the observations of all the robots, updating the trajectories and the map jointly ([31, 9, 12, 15]). In this case, the robots will have a global notion of the space, what facilitates the map exploration tasks. Nevertheless, the problem is that the initial relative position of the robots should be known, which is something that may not be possible in practice.

2. Solutions in which each robot estimates an own individual map using its observations independently ([18, 36]). In this case, new observations should only be compared with a limited number of landmarks in the local maps. Additionally, the construction of the local maps can be carried out even if the relative poses of the robots are not known. This is an advantage over the previous case. However, the map fusion step is troubled since the data association should be solved between the local maps.

In this paper, we focus on the latter approach. i.e., the robots start from different positions and build local maps independently. Then, the fusion of these local maps may be required. As a consequence, the trasformation between the different reference systems should be known. In this situation, most approaches try to find the relative position of the robots. In this sense, the easiest case can be seen in [31], where the relative position of the robots is supposed to be known. Nevertheless, more difficult approaches are [18] and [36]. In these cases, the robots try to establish a meeting point in order to measure their relative positions. In many approaches the transformation between maps is performed with the matching of landmarks [29].

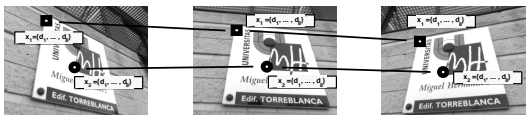


Figure 1: Tracking of Harris points described by u-SURF

3 Visual Landmarks

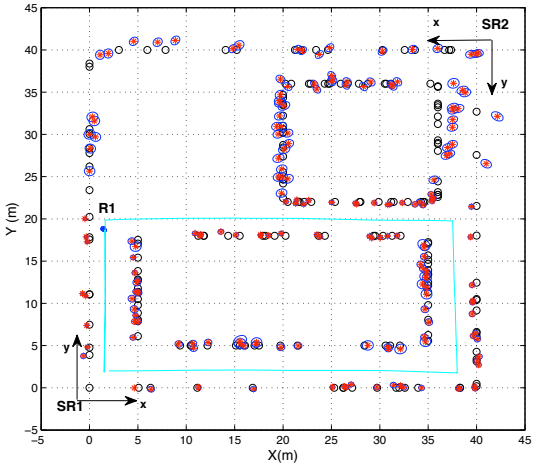
As mentioned before, in our case, the robots build their maps using visual information from the environment. To do this, they use a stereo head mounted on them. Most of maps using visual information are *landmark-based*. Those landmarks represent the location of a set of points from the environment with respect to a global reference frame. The main advantage of this representation is the compactness.

Since we use stereo vision, the landmarks represent the 3D position of the points. Mainly, two steps must be distinguished in the selection of visual landmarks. The first step involves the detection of interest points in the environment. The detection should be as stable as possible, since the points of the environment are observed from different viewpoints. Then, at a second step the interest points are described by a feature vector which is computed using local image information. This descriptor is used in the data association problem, i.e., when the robot has to decide whether the current observation corresponds to one of the landmarks in the map or to a new one. Different detectors and descriptors have been used for mapping and localization using monocular or stereo vision, such as SIFT [20, 13, 33], the Harris corner detector [8, 16], Harris-Laplace [17] or SURF [26].

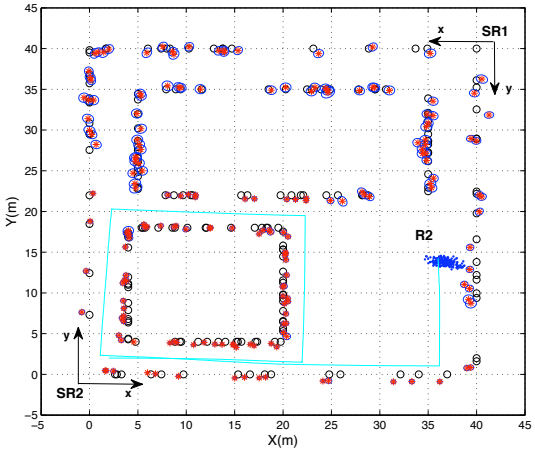
In a prior work, we performed a comparative study in order to find the most suitable combination detector-descriptor in the visual SLAM context [23, 2, 11]. As a result, we obtained that the best feature extractor was the Harris Corner detector combined with the u-SURF descriptor. This detector/descriptor proved to be the most suitable for visual SLAM. The u-SURF descriptor is not rotationally invariant [5]. However, this is not a problem in our case since the stereo camera is fixed on the robot and it only performs movements in a 2D plane. In a different situation with more DOGs, the SURF descriptor would work properly.

4 Map building

In this work, we use Pioneer-P3AT robots, provided with a laser sensor and a STH-MDCS2 stereo head



(a)



(b)

Figure 2: Example of map building using FastSLAM. Two robots share the same space ($R1$ and $R2$), but the map building is performed independently.

from Videre Design. This stereo camera is used to extract visual information from the environment. Concretely, we use the Harris corner detector [14] to obtain distinctive points of the scene. Moreover, these points are characterized by a visual descriptor known as U-SURF [5]. The selection of this combination detector/descriptor is the result of a previous work [11].

As mentioned before, in this paper the SLAM problem is solved using the FastSLAM algorithm . The main idea of the FastSLAM algorithm is that the SLAM problem can be separated into two main subproblems: the estimate of the trayectory of the robot and the estimate of the map [25]. This can be expressed as:

$$p(x^t, L|z^t, u^t, c^t) = p(x^t|z^t, u^t, c^t) \prod_{k=1}^{N} p(l_k|x^t, z^t, u^t, c^t) \quad (1)$$

This equation states that the SLAM posterior is decomposed into two parts: the estimate of the robot path and N independent estimators of the landmark positions, each conditioned to the path estimate. We approximate $p(x^t|z^t, u^t, c^t)$ by means of a set of M particles. Thus, each particle has N independent landmark estimators (implemented as EKFs), one for each landmark. Each particle is therefore defined as:

$$S_t^{[m]} = \{x^{t,[m]}, \mu_{t,1}^{[m]}, \Sigma_{t,1}^{[m]}, d_1^{[m]}, \ldots, mu_{t,N}^{[m]}, \Sigma_{t,N}^{[m]}, d_N^{[m]}\}, \quad (2)$$

where $\mu_{t,k}^{[m]}$ is the best estimation at time t for the position of landmark l_k based on the path of the particle m and $\Sigma_{t,k}^{[m]}$ the associated covariance matrix. The visual descriptor associated to the landmark j is represented by $d_j^{[m]}$. The particle set $S_t = \{S_t^{[1]}, S_t^{[2]}, \ldots, S_t^{[M]}\}$ is calculated incrementally from the set S_{t-1} in time $t-1$ and the control u_t.

This algorithm can be summarized in the following steps:

1. New particle set generation. In a first step, a new set of particles representing the location of the robot are obtained from the previous set. That is to say, these particles evolve taking into account the previous position each particle x_{t-1} and the movement performed by the robot u_t. For each particle m, this can be expressed as:

$$x_t^{[m]} \sim p(x_t|x_{t-1}, u_t) \quad (3)$$

These particle follow a gaussian distribution. At the initial position of the robot, all particles are

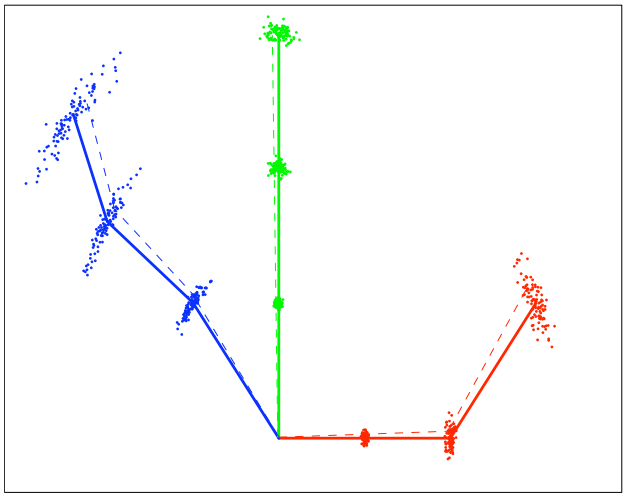


Figure 3: The dispersion of the particle set grows as the robot moves, representing the uncertainty in the robot's pose. This figure shows the evolution of the particle set along the path of three different robots.

concentrated in the same location. Afterwards, as long as the robot moves, the uncertainty on its pose grows and therefore the dispersion of the particles is higher. This uncertainty can be reduced if, for example, the robot revisits an area. Both situations are shown in figure 4. Fig. 4(a) presents the moment in which the robot closes a loop and thus it reobserves landmarks previously integrated in the map. In this case, the uncertainty of the pose of the robot is small so the particles are concentrated. On the contrary, in Fig. 4(b) presents the situation in which the robot is performing several movements in a new area. In this case the set of particles is more dispersed indicating that the uncertainty is higher.

2. Landmark estimation.

   The next step consist on updating the estimate of the landmarks in the map. When a robot performs an observation, it identifies whether the landmark is observed for the first time or, on the contrary, it corresponds to a landmark previously integrated in the map. This problem is known as data association. In this step we concentrate on how the estimate of the landmarks is updated based on the pose of the robot, having made the observation $o_t = \{z_t, d_t\}$ (z_t represents the coordinates of the point detected and d_t the descriptor) with data association c_t. The update of each landmark θ_{c_t} is performed independently for each particle by means of the EKF (Extended Kalman Filter) equations as detailed here:

$$\hat{z}_t = g(x_t^{[m]}, \mu_{c_t,t-1}^{[m]}) \quad (4)$$

$$G_{l_{c_t}} = \nabla_{l_{c_t}} g(x_t, l_{c_t})_{x_t=x_t^{[m]}; l_{c_t}=\mu_{c_t,t-1}^{[m]}} \quad (5)$$

$$Z_{c_t,t} = G_{l_{c_t}} \Sigma_{c_t,t-1}^{[m]} G_{l_{c_t}}^T + R_t \quad (6)$$

$$K_t = \Sigma_{c_t,t-1}^{[m]} G_{l_{c_t}}^T Z_{c_t,t}^{-1} \quad (7)$$

$$\mu_{c_t,t}^{[m]} = \mu_{c_t,t-1}^{[m]} + K_t(z_t - \hat{z}_t) \quad (8)$$

$$\Sigma_{c_t,t}^{[m]} = (I - K_t G_{l_{c_t}}) \Sigma_{c_t,t-1}^{[m]} \quad (9)$$

   where $\hat{z}_t$ is the prediction for the current measurement z_t assuming that it has been associated with landmark c_t in the map. The observation model $g(x_t, l_{c_t})$ is linearly approximated by the Jacobian matrix $G_{l_{c_t}}$. It is assumed here that the noise in the observation is Gaussian and can be modeled with the covariance matrix R_t. Equation (8) represents the update of the estimate of the landmark c_t: $\mu_{c_t,t-1}^{[m]}$ based on the innovation $z = (z_t - \hat{z}_t)$. Finally, Equation (9) updates the covariance matrix $\Sigma_{c_t,t}^{[m]}$, which is associated to the m particle and the landmark c_t. Note that we implicitly assume that the observation z_t corresponds to the landmark l_{c_t} in the map.

3. Assigning a weight to each particle.

   Next, a weight is given to each particle based on the quality of the correspondence between the observations performed and its associated map. This weight is computed as:

$$\omega_t^{[m]} = \frac{1}{\sqrt{|2\pi Z_{c_t}|}} e^{\{-\frac{1}{2}(v_t - \hat{v}_{t,c_t})^T [Z_{c_t}]^{-1} (v_t - \hat{v}_{t,c_t})\}} \quad (10)$$

   The particles with the highest values of the weights, will be the most probable particles.

4. Importance reampling.

   Finally, a resampling process is made in order to keep the particles with high weights. Those with lower weight values are replaced by other with higher ones. This step is not performed at each iteration of the FastSLAM algorithm, since this would reduce the particles variety, affecting negatively to the results.

Figure 2 shows an example of the map building using the FastSLAM algorithm. Two robots share the same scenario although they do not have any knowledge about the other robot's existence. Each robot perfoms an independent particle filter. In the figure, we

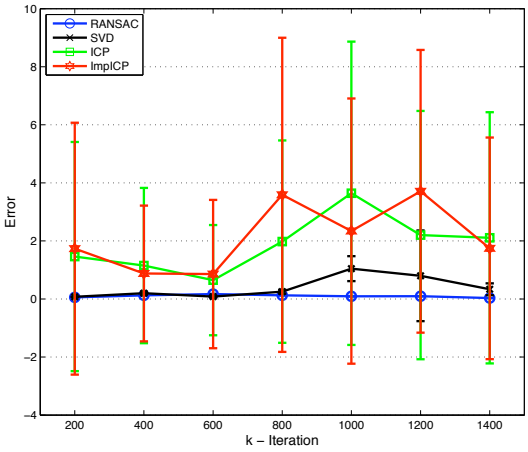


Figure 4: Traslation error.

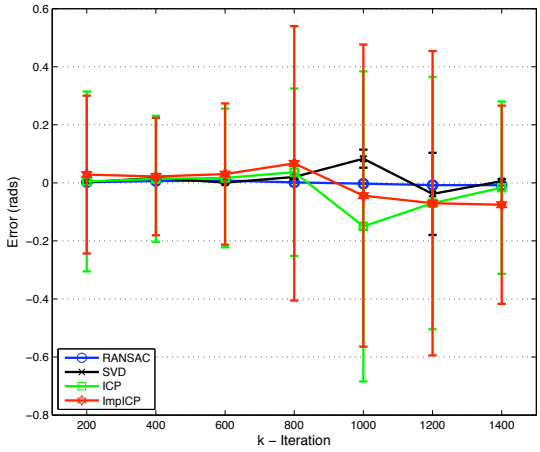


Figure 5: Rotation error.

can see the reference system of each robot (SR_1 and SR_1). Figure 2(b) presents the same scene of Figure 2 (a), but rotated as SR_2. The reference systems are located at the (0,0,0) position of the respective robots. In the figure, we apreciate how the pose of each robot is represented by a particle set. $R1$ (Figure 2(a)) has a lower uncertainty in the pose since the robot has already closed a loop (revisits an area). On the contrary, in Figure 2(b), we observe that the uncertainty in the pose of $R2$ is higher since the particle set is more disperse. Additionally, the path of the robots is also represented. For clarity reasons, we present only the path of the most probable particle, which is the best estimate at that moment. Regarding the map estimated, it can be observed that the estimate of the landmarks has more or less uncertainty depending on how many times are these landmarks seen by the robots or the distance from which they are observed. The uncertainty is represented by an ellipse.

In the experiments presented in this paper, the map alignment is evaluated at different stages of the SLAM process. These experiments have been carried out using 200 particles per robot.

5 Map Aligment

This section studies the aligment of landmark-based maps. Concretely, the maps built are made of visual landmarks. Aligning two maps means establishing a common reference system for these maps by computing three aligning parameters: t_x, t_y and θ. This is done by computing the transformation between the reference systems of the different local maps.

In this framework, our aim is to find a suitable method that allows us to align this kind of maps. In order to do this, we have performed an evaluation of a set of aligning methods that are enumerated below:

Algorithm 1 Computation of T, given m and m'

1: $[u, d, v] = svd(m')$
2: $z = u^T m$
3: $sv = diag(d)$
4: $z_1 = z(1:n)$ {n is the number of eigenvalues (not equal to 0) in sv.}
5: $w = z_1./sv$
6: $T = (v * w)^T$

1. RANSAC (Random Sample Consensus). This algorithm have been already used in map alignment in [29]. It is an iterative algorithm in which the first step is to identify the correspondent landmarks between both maps. Then two pairs of corresponcences are selected at random and an initial estimate of the alignment is computed. This proces is repeated a number of times. At each time, the set of correspondences that support the solution obtained. The alignment computed is that one with the higher number of supports.

2. SVD (Singular Value Decomposition) [1, 28]. This algorithm also begins with a list of correspondences between the two maps. Then the alignment is computed as it is shown in Algorithm 1.

3. ICP (Iterative Closest Point) [6, 35]. This is an iterative algorithm in which the objective is to minimize the following expression: $\|T.m' - m\|$, where m and m' are the correspondences and T is the transformation matrix constituted by the

three alignment parameters as shown here:

$$T = \begin{pmatrix} \cos\theta & \sin\theta & 0 & t_x \\ -\sin\theta & \cos\theta & 0 & t_y \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (11)$$

4. ImpICP(Improved ICP) [4, 3]. The ImpICP method is a modification of ICP implemented *ad hoc* in order to increase the probability of obtaining a good estimate.

The rest of methods have been already applied to map alignment or point registration [21, 24] . A more detailed explanation of the functioning of these methods can be seen in [3, 4]. Basically, all these methods establish correspondences between the landmaks of two local maps, based on the descriptor similarity. Then, given this set of correspondences, an estimate of the alignment is computed.

It is noticeable that these methods obtain only a first estimate of the aligning parameters. The set of correspondences and this estimate are used as the input of a *least squares minimization* that eliminates outliers and obtains the final solution [27].

Moreover, these aligning methods were evaluated not only qualitatively but also in terms of their computational efficiency. In Figure 6, a comparison of the computational time of the aligning methods. In this figure, we present the time that it takes to obtain the aligning parameters (seconds) *vs*. different number of correspondent points between the local maps. Logically, the time is higher as the common part between the maps is bigger. It can be observed that the computational time of the different aligning methods is very similar, so it can be deduced that this is not a determinant factor in order to select one of these methods as the most suitable to align visual landmark-based maps.

In these experiments, the local maps have been built by means of the FastSLAM algorithm. This algorithm is performed in several iterations. Since the aim of this study is to observe the behaviour of the aligning methods at different stages of the SLAM process, we obtain the most probable map at each selected iteration. The most probable map is the map of the most problable particle at that specific moment. Then, given two maps, the alignment is carried out by each aligning method. Finally, the solution is evaluated as an error measure computing the Euclidean distance between that solution and a *ground truth*. This *ground truth* is a measure of the real relative initial position of the robots.

Figures 4 and 5 show the results obtained after comparing the aligning methods previously mentioned. They present the error in the estimate of the

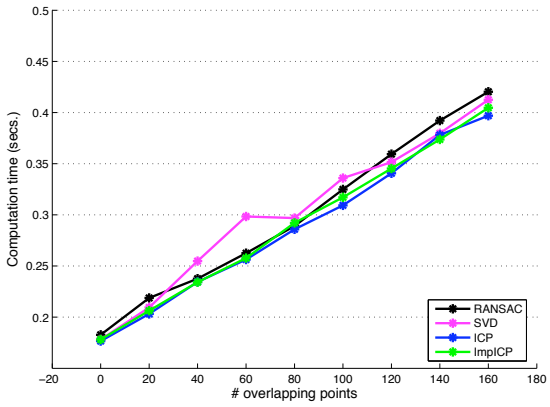


Figure 6: Computational time vs. number of overlapping points between the local maps.

aligning parameters *vs*. the $k - Iteration$ of the FastSLAM algorithm. Figure 4 shows the traslation error (in meters), i.e., in the estimate of tx and ty. Then, figure 5 presents the rotation error expressed in radians (estimate of θ).

As the iteration of FastSLAM is higher, i. e., when the number of landmarks in the maps grows, two situations may arise. On the one hand, it is probable that the overlapping part between the local maps is bigger, i.e., there will be more correspondences between the maps we want to align. In this situation, the estimate of the aligning parameters will be better. This fact is visible in the results obtained. Particularly, it can be seen in Figure 4 how the error of the solutions obtained by ICP and ImpICP decreases from $k - Iteration = 200$ till $k - Iteration = 600$. On the other hand, having more landmarks does not mean necesarily having more correspondences. For this reason, when the size of the maps grows, it can happen that the non-overlapping parts are bigger. This fact adds complexity to the search of correspondences (preliminary step of the aligning methods to compute the alignment). In this cases, the aligning methods are requested to be specially robust to false correspondences. In Figures 4 and 5 it can be observed that the error obtained is bigger around $k - Iteration = 1000$. Nevertheless, it is worth noting that RANSAC is invariant to the situations described. Moreover, it obtains a quite accurate estimate of the alignment, since the error values are very close to zero. RANSAC is therefore an aligning method robust to the percentage of common landmarks and is able to obtain low error results. Regarding the rest of methods, SVD obtain acceptable solutions although not so accurate as RANSAC. ICP and ImpICP do not obtain good results, since obtain errors close to 4 meters in the estimate of the traslation and close to -0.2 radians in rota-

tion (ICP). Furthermore, they present results with high variance, what denotes some randomness in the estimate of the alingment.

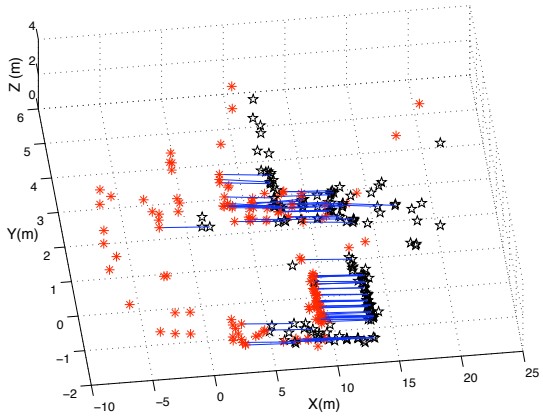


Figure 7: Correspondences established between two maps to be aligned.

Figure 7 shows an example of two maps (represented by asterisks and stars) as the typical used to be aligned is these experiments. These are 3D visual maps that initially have different reference systems. In the figure, it can be noticed that a set of common landmarks (correspondences) have been identified between the maps. These correspondences are used to compute the aligment between the maps.

6 Map merging

Once the aligment is performed, the local maps have the same reference system. However, in order to obtain a unique global map, these local maps have to be merged. Figure 8 presents the situation in which the same point of the scene (θ) has been observed by two robots ($ROBOT1$ and $ROBOT2$) from different positions. This point is incorporated by each robot as a landmark in its respective local map. Particularly, the landmark is added as L_i and L_j respectively, as shown in Figure 8. Logically, the same landmark in different local maps will have different uncertainty (Σ_i and Σ_j). This uncertainty is represented in Figure 8 as an ellipse and depends on several factors, such as the distance between the robot and the landmark when it is observed, the uncertainty on the pose of the robot and the fact that this landmark can be reobserved during the SLAM process. Those factors affect the magnitude of the uncertainty in the estimate of the landmarks represented by the size of the ellipse.

It is noticeable that when merging two local maps, the uncertainty of the landmarks have to be taken into

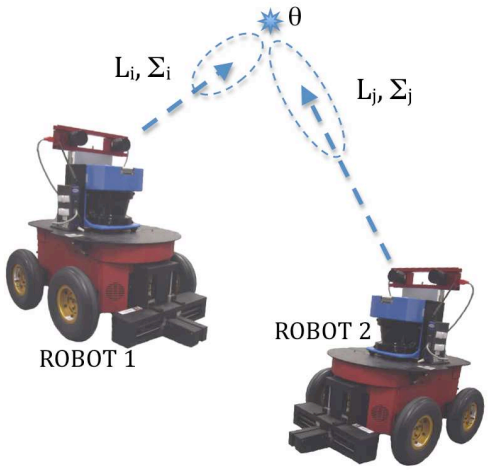


Figure 8: The same landmark θ has been observed by two different robots and integrated in their respective maps as L_i and L_j.

account. For this purpose, our proposal in this paper is a Multivariable Stationary Kalman filter. Given two maps (1 and 2), the fused map can be obtained by means of the following formulation:

$$K_{\{m\}} = \Sigma_{i\{m\}} \cdot (\Sigma_{i\{m\}} + \Sigma_{j\{m\}})^{-1} \quad (12)$$

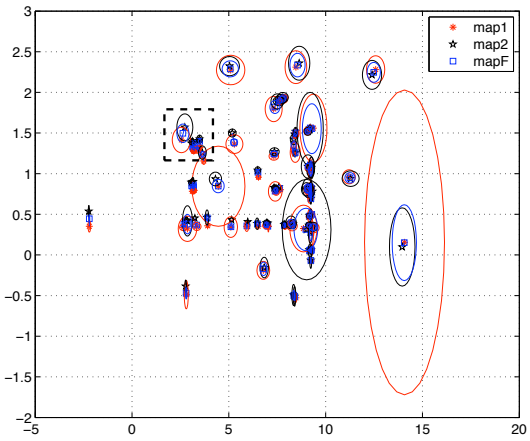
$$L_{F\{m\}} = L_{i\{m\}} + K_{\{m\}} \cdot (L_{i\{m\}} - L_{j\{m\}}) \quad (13)$$

$$\Sigma_{F\{m\}} = (I - K_{\{m\}}) \cdot \Sigma_{i\{m\}} \quad (14)$$

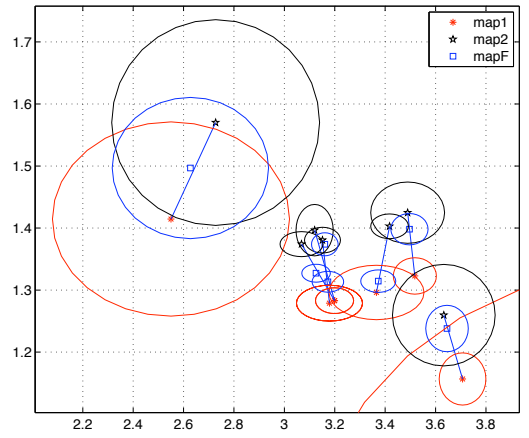
where m is an index ($m \in \{1, M\}$, M: number of correspondences between the local maps) that denotes each pair of correspondences between the maps (in this case, i and j). L_i, L_j and L_F are the 3D coordinates of the landmaks in map_i, map_j and the fused map_F respectively. It is noticeable that map_i and map_j have been already aligned and therefore the landmarks are expressed in the same reference system. Then, Σ_i, Σ_j and Σ_F represent, by means of a 3×3 covariance matrix, the uncertainty of the landmarks belonging to map_i, map_j and map_F. It is remarkable that the aligment is not only applied to the coordinates of the landmarks, but also to the uncertainty ellipse. This is done by means of a rotation matrix (R) as shown below:

$$\Sigma_j = R^T \cdot \Sigma_{j0} \cdot R \quad (15)$$

$$R = \begin{pmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad (16)$$



(a)



(b)

Figure 9: Results of map merging (2D view). (a) Presents correspondences of map_1 and map_2 aligned and fused into map_F. Error ellipses are also represented. (b) Zoom of the black rectangle drawn in (a). The fused landmarks (map_F) present a lower uncertainty (smaller ellipses).

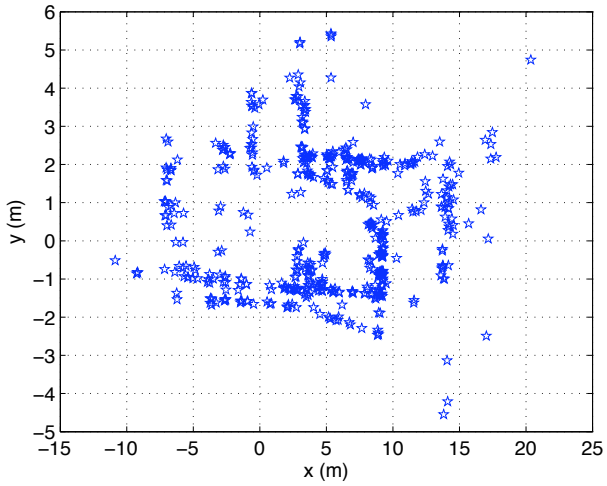


Figure 10: Maps of Figure 7 are merged into a global one.

where Σ_{j0} is the covariance of map_j before the alignment.

In Figure 9 a real example of map merging is shown. Concretely, Figure 9(a) presents a set of landmarks identified as correspondences between two maps (1 and 2). In the figure, these maps have been already aligned so the correspondent landmarks almost overlap. Moreover, the resulting fused map (map_F) is also represented. Finally, the uncertainty in the estimate of the landmarks is represented by ellipses. For clarity reasons, a small area of this figure has been enlarged. Thus, the dashed rectangle is broaden to Figure 9(b). In this case, the correspondences can be seen connected by a line. Landmarks belonging to map_1 are represented by an asterisk and those of map_2 are represented by a star. Finally, the landmarks of the obtained map_F are represented by squares. As shown in Figure 9(b), the new landmaks, i.e., the landmarks of the fused map have lower uncertainty values since the uncertainty ellipses are smaller. Finally, Figure 10 shows a 2D view of a fused map, which is the result of merging the maps of Figure 7.

7 Conclusion

The approach proposed here consists in maintaining independent particle filters in a multi-robot platform. In this case the relative positions of the robots are not needed *a priori*, since the robot initate their map building task without notion of other robots' positions or observations. Furthermore, it is less computationally expensive than the case in which the map and trajectories is performed jointly. In this case, the local maps are smaller and each filter only computes the path of a

single robot.

In a next step, we consider the situation in which these robots want to fuse their local maps into a single one. We therefore study the map fusion problem by dividing it into an alignment problem and a merging problem. In the first case a comparison of several aligning methods was made. As a result, we concluded that RANSAC is the most suitable aligning method for this kind of maps, i.e., visual landmark-based maps. The experiments also show that the global map obtained presents less uncertainty than the original local maps, thanks to the Multivariable Stationary Kalman filter. The results obtained regarding the map alignment and fusion problem are useful for any application using landmark-based maps.

Acknowledgements: The research has been supported by the Spanish Ministry of Science and Innovation under projects DPI2004-07433- C02-01 and CICYT DPI2007-61197 and by the Generalitat Valenciana (grant BFPI/2007/096).

References:

[1] K.S. Arun, T.S. Huang, and S.D. Blostein. Least square fitting of two 3d sets. In *IEEE Transactions on Patterns Analysis and Machine Intelligence. vol. PAMI-9 no. 5, pp.698-700*, 1987.

[2] M. Ballesta, A. Gil, O. Martínez Mozos, and O. Reinoso. Local descriptors for visual SLAM. In *Workshop on Robotics and Mathematics (ROBOMAT07), Portugal*, 2007.

[3] M. Ballesta, A. Gil, O. Reinoso, M. Julia, and L.M. Jimenez. Alignment of visual maps in multirobot fastslam. In *8th WSEAS International Conference on Computational Intelligence, Man-machine Systems and Cybernetics (CIMMACS)*, pages 92–97, 2009.

[4] M. Ballesta, A. Gil, O. Reinoso, M. Julia, and L.M. Jimenez. Multi-robot map alignment in visual slam. *WSEAS TRANSACTIONS on SYTEMS*, 9:213–222, 2010.

[5] H. Bay, T. Tuytelaars, and L. Van Gool. Surf: Speeded-up robust features. In *Proc. of the 9th European Conference on Computer Vision*, 2006.

[6] P.J. Besl and N. McKay. A method for registration of 3-d shapes. In *IEEE Transactions on Patterns Analysis and Machine Intelligence. vol. PAMI-14 no. 2, pp. 239-256*, 1992.

[7] A. Cumani, S. Denasi, A. Guiducci, and G. Quaglia. Integrating monocular vision and odometry for slam. In *WSEAS Transactions on Computers*, volume 3, pages 625–630, 2004.

[8] Andrew J. Davison and David W. Murray. Simultaneous localisation and map-building using active vision. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2002.

[9] John W. Fenwick, Paul N. Newman, and John J. Leornard. Cooperative concurrent mapping and localization. In *Proc. of the 2002 IEEE International Conference on Intelligent Robotics and Automation, pp.1810-1817*, 2002.

[10] R. Garcia-Garcia, M.A. Sotelo, I. Parra, D. Fernandez, Naranjo J.E., and M. Gavilan. 3d visual odometry for road vehicles. *Journal of Intelligent and Robotic Systems*, 51(1):113–134, 2008.

[11] A. Gil, O. Martinez Mozos, M. Ballesta, and O. Reinoso. A comparative evaluation of interest point detectors and local descriptors for visual slam. *Machine Vision and Aplications*, pages 1432–1769, 2009.

[12] A. Gil, O. Reinoso, M. Ballesta, and M. Juliá. Multi-robot visual slam using a rao-blackwellized particle lter. *Robotics and Autonomous Systems Journal*, pages 68–80, 2010.

[13] A. Gil, O. Reinoso, W. Burgard, C. Stachniss, and O. Martínez Mozos. Improving data association in rao-blackwellized visual SLAM. In *IEEE/RSJ Int. Conf. on Intelligent Robots & Systems*, 2006.

[14] C. G. Harris and M. Stephens. A combined corner and edge detector. In *Alvey Vision Conference*, 1998.

[15] A. Howard. Multi-robot simultaneous localization and mapping using particle filters. In *The International Journal of Robotics Research, Vol. 25, No. 12, 1243-1256*, 2006.

[16] Emmanuel Hygounenc, Il-Kyun Jung, Philippe Souères, and Simon Lacroix. The autonomous blimp project of laas-cnrs: Achievements in flight control and terrain mapping. *International Journal of Robotics Research*, 23(4–5), 2004.

[17] Patric Jensfelt, Danica Kragic, John Folkesson, and Mårten Björkman. A framework for vision based bearing only 3D SLAM. In *IEEE Int. Conf. on Robotics & Automation*, 2006.

[18] K. Konolige, D. Fox, B. Limketkai, J. Ko, and B. Stewart. Map merging for distributed robot navigation. In *Proc. of the 2003 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2003.

[19] J.J. Leonard and H.F. Durrant-Whyte. Mobile robot localization by tracking geometric beacons. *IEEE Transactions on Robotics and Automation*, 7(4), 1991.

[20] J. Little, S. Se, and D.G. Lowe. Global localization using distinctive visual features. In *IEEE/RSJ Int. Conf. on Intelligent Robots & Systems*, 2002.

[21] D. Maier, J. Hessler, and R. MŁnner. Fast and accurate closest point search on triangulated surfaces and its application to head motion estimation. In *3rd WSEAS International Conference on SIGNAL, SPEECH and IMAGE PROCESSING*, 2003.

[22] A. Mancini, A. Cesetti, A. Iuale, E. Frontoni, P. Zingaretti, and S. Longui. A framework for simulation and testing of uavs in cooperative scenarios. *Journal of Intelligent and Robotic Systems*, 54(1-3):307–329, 2009.

[23] O. Martínez Mozos, A. Gil, M. Ballesta, and O. Reinoso. Interest point detectors for visual slam. In *Proc. of the XII Conference of the Spanish Association for Artificial Intelligence (CAEPIA), Salamanca, Spain*, 2007.

[24] N. E. Mastorakis. The singular value decomposition (svd) in tensors (multidimensional arrays) as an optimization problem. solution via genetic algorithms and method of nelder-mead. In *6th WSEAS International Conference on Systems Theory and Scientific Computation*, pages 7–13, 2006.

[25] M. Montemerlo, S. Thrun, D. Koller, and B. Wegbreit. Fastslam: A factored solution to simultaneous localization and mapping. In *Proc. of the National Conference on Artificial Intelligence (AAAI), pp. 593598. Edmonton, Canada*, 2002.

[26] A. C. Murillo, J. J. Guerrero, and C. Sagüés. Surf features for efficient robot localization with omnidirectional images. In *IEEE Int. Conf. on Robotics & Automation*, 2007.

[27] S.T. Pfister, K. L. Kreichbaum, S.I. Roumeliotis, and J.W. Burdirck. Weighted range sensor matching algorithms for mobile robot displacement estimation. In *IEEE International Conference on Robotics and Automation, Washington D.C., pp 1667-74*, 2002.

[28] J. Rieger. On the classification of views of piecewise smooth objects. In *Image and Vision Computing, vol. 5, no. 2, pp. 91-97*, 1987.

[29] S. Se, D. Lowe, and J.J. Little. Vision-based global localization and mapping for mobile robots. In *IEEE Transactions on robotics, vol.21, no.3*, 2005.

[30] S. Takahashi, I. R. Khan, M. Okuda, and Masaaki Ikehara. Real time 3d avatar transmission using cylinder mapping. *WSEAS Transactions on Electronics, Special Issue on Real-Time Applications With 3D Sensors*, 5, 2008.

[31] S. Thrun. A probabilistic online mapping algorithm for teams of mobile robots. In *Int. Journal of Robotics Research, 20(5), pp. 335363*, 2001.

[32] R. Triebel and W. Burgard. Improving simultaneous mapping and localization. In *Proc. of the National Conference on Artificial Intelligence (AAAI)*, 2005.

[33] J. Valls Miro, W. Zhou, and G. Dissanayake. Towards vision based navigation in large indoor environments. In *IEEE/RSJ Int. Conf. on Intelligent Robots & Systems*, 2006.

[34] O. Wijk and H. I. Christensen. Localization and navigation of a mobile robot using natural point landmarkd extracted from sonar data. In *Robotics and Autonomous Systems, 1(31), pp. 3142*, 2000.

[35] Z. Zhang. On local matching of free-form curves. In *Proc. of BMVC, pp. 347-356*, 1992.

[36] Xun S. Zhou and Sergios I. Roumeliotis. Multi-robot slam with unknown initial corrspondence: The robot rendezvous case. In *Proc. of the 2006 IEEE/RSJ International Conference on Intelligent Robots and Systems, Beijing, China, pp. 1785-1792*, 2006.