

Cooperative localization by fusing vision-based bearing measurements and motion

Luis Montesano¹José Gaspar²José Santos-Victor²Luis Montano¹

¹IA, Dpto. de Informática e Ing. de Sistemas
Universidad de Zaragoza, Spain
{montesano,montano}@unizar.es

²ISR, Instituto Superior Tecnico
Lisboa, Portugal
{jag,jasv}@isr.ist.utl.pt

Abstract—This paper presents a method to cooperatively localize pairs of robots fusing bearing-only information provided by cameras and the motion of the vehicles. The algorithm uses the robots as landmarks to estimate their relative location. Bearings are the simplest measurements directly obtained from the cameras, as opposed to measuring depths which would require knowledge or reconstruction of the world structure.

We present the general recursive Bayes estimator and three different implementations based on an extended Kalman filter, a particle filter and a combination of both techniques. We have compared the performance of the different implementations using real data acquired with two platforms equipped with omnidirectional cameras and simulated data.

I. INTRODUCTION

In the last years cooperative robotics has received considerable attention. Teams of robots are able to overcome the limitations of single robots and allow to attack more difficult tasks. Furthermore, they increase the degree of autonomy and robustness by introducing redundancy. However, the use of teams of robots increases the complexity of the system. New challenges appear providing new research areas. In this context cooperative localization is considered one of the basic capabilities required for autonomous operation of teams of robots.

In this paper we address the problem of cooperatively localizing two robots using vision-based bearing-only measurements and the motion of the robots. Exteroceptive sensors, such as vision, allow relating the independent referential frames created and maintained by the robots' proprioceptive sensors (e.g. odometry) [17]. Omnidirectional cameras give bearing (azimuthal) locations of robots in a 360° field-of-view. As compared to vision-based SLAM ([3], [18]), stereo matching and reconstruction ([4]), ground-plane based navigation, or other depth reconstruction techniques, extracting bearings is a simple process, close to obtaining raw data. Advantageously, the tracked-robot is considered a single point, and therefore there are no assumptions of rigidity or smoothness of a multi-point structure.

As the robots are moving the bearing measurements do not contain enough information to estimate their relative location. We propose to use the motion of the vehicles (odometry readings or motion commands) together with the bearing observations. Combining both types of information allows us to estimate the initial relative location of the

platforms and keep track of it. In order to take into account heterogeneous pairs of robots and possible hardware failures we have implemented our localization algorithm for two different configurations. In the first one only one robot has detection capabilities while in the second both robots are equipped with sensors providing the bearing of the other robot. For each configuration we have evaluated three implementations of the classical Bayes filter: an extended Kalman filter, a particle filter and a combination of both techniques. We have performed several tests in our laboratory to validate the method. The results suggest that we can fuse the motion of the platforms with the bearing measurements to estimate their relative location.

This paper is organized as follows. Next Section presents the related work. Section III describes the problem. In Section IV and V we derive the Bayesian estimator for our problem and we describe our implementation. The experimental results are presented in Section VI.

II. RELATED WORK

In a computer vision framework the relative localization of two robots corresponds to an object-pose estimation problem. Seminal work on object-pose estimation assumed known geometrical object-structures useful both for object detection and tracking [10], [9]. The most recent works show that an object can be tracked, without knowing its structure, using other object-features such as image corners, contours, patterns, color patches, etc ([13], [2], [15]). Many of the proposed tracking methods do not provide pose estimation, due to the fact that feature models do not contain structure information. However they obtain the location of the object within the image. In the present work, we extract only bearings from the images and combine them with the motion of both platforms to estimate their relative localization.

On the other hand, cooperative localization has been an active research area during the last years in the robotic community. There exist different approaches depending on several aspects as the information used (map availability, static/dynamic features) or the multi-robot architecture (communication capabilities, centralized/decentralized). Related to this paper several authors have used the robots as landmarks ([17], [8], [12]). In the absence of static references these methods estimate the relative pose among the robots instead of their location on a

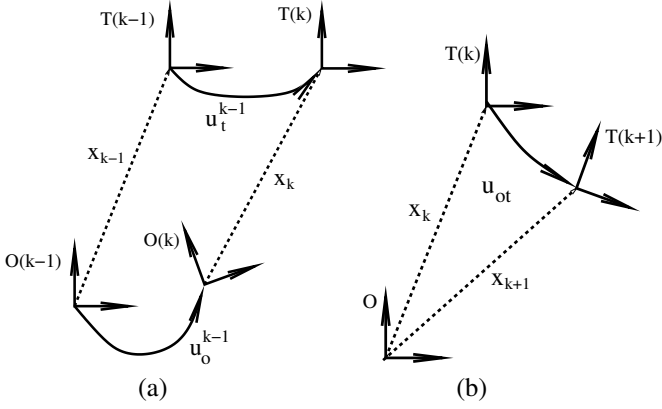


Fig. 1. Interpretation of the relative location between two moving platforms. (a) world fixed reference system (b) robocentric approach.

global reference system. However, all these algorithms use range-bearing sensors that provide an accurate estimation of their relative positions.

Some authors have used bearing-only measurements to estimate the location of the robots or the objects around them [19], [16]. The proposed algorithms usually use probabilistic techniques to fuse the information obtained at the same time by different sources with known uncertain locations. Bearing-only information has also been studied to track constant velocity targets using Kalman filters [1] or particle filters [14]. Our method, instead of restricting the motion of the robots with a known model, uses the measurements of their displacements to turn the problem observable.

III. PROBLEM DESCRIPTION

We are interested on estimating the relative location \mathbf{x} of two moving platforms. As we do not have any static reference we use a robocentric approach [8]. We choose one of the platforms as the reference system, the observer, and compute the relative pose of the other one, the target, with respect to it. In this reference system the observer remains static while the target motion is the composition of the motion of both platforms. Figure 1(a) depicts the relations between the locations of two moving platforms. The observer and target motion is \mathbf{u}_o and \mathbf{u}_t respectively. When the reference system moves with the observer (Figure 1(b)) the final location of the target is the result of the combination of the movements of both platforms \mathbf{u}_{ot} .

According to the previous description our problem is equivalent to estimating the position of a moving target from a stationary platform. Bearing-only measurements of a moving target do not contain enough information to estimate its position [1]. Let $\mathbf{p}_t = (x_t, y_t)$ be the position of the target and $\mathbf{p}_o = (x_o, y_o)$ the position of the observer. We will denote $\mathbf{p}_o(k)$ to the position of the target at time step k . The observations are bearings with respect to a reference direction of the target at different points in time,

$$z^k = \tan^{-1} \frac{y_t(k) - y_o(k)}{x_t(k) - x_o(k)} + w^k$$

where w^k is the measurement noise at time k . Due to the relative motion of the target, which is a combination of its own motion and that of the observer, there exist infinite possible locations of the platform satisfying the measurement equation.

One, therefore, needs to incorporate some more information to be able to compute the relative pose of the platforms. If one knows the relative motion of the target, its location can be estimated using three measurements and triangulation techniques. The relative motion restricts the set of possible trajectories to a single one. In order to incorporate the motion of the target one must estimate not only its position but the whole location including the orientation $\mathbf{x}_t = (x_t, y_t, \theta_t)$. There exist some degenerated cases due the combination of the motion of the vehicles (parallel trajectories, moving along the line joining both platforms or just rotating).

Finally, the method has to deal with the uncertainties and noises in the system. In the next Section we present a Bayesian estimation technique to estimate the relative pose of two moving vehicles using bearing measurements and the trajectories of both vehicles.

IV. RELATIVE POSE ESTIMATION

We assume that the robots are able to measure their own displacement and at least one of them is equipped with sensors that provide bearing measurements z of the position of the other robot. Let \mathbf{u}^k be the motion of a robot between time $k-1$ and k and z^k the bearing obtained at time k . Let r_o be the observer robot. We will denote z_o^k the observations and \mathbf{u}_o^k the odometry readings obtained by r_o at time k . $Z_o^k = \{z_o^1, \dots, z_o^k\}$ and $U_o^k = \{\mathbf{u}_o^1, \dots, \mathbf{u}_o^k\}$ represent the observations and odometry readings of the observer up to time k . The target observations and motion measurements up to time k are $Z_t^k = \{z_t^1, \dots, z_t^k\}$ and $U_t^k = \{\mathbf{u}_t^1, \dots, \mathbf{u}_t^k\}$ respectively. Our objective is to estimate the relative location $\mathbf{x} = (x \ y \ \theta)$ of robot r_t , the target, with respect to robot r_o based on the information available up to time k . Thus, the posterior distribution we are estimating is $p(\mathbf{x}_k | Z_o^k, U_o^k, Z_t^k, U_t^k)$ and the corresponding recursive Bayes filter is,

$$p(\mathbf{x}_k | Z_o^k, U_o^k, Z_t^k, U_t^k) \propto p(z_o^k | \mathbf{x}_k) p(z_t^k | \mathbf{x}_k) p(\mathbf{x}_k | \mathbf{x}_{k-1}, Z_o^{k-1}, U_o^k, Z_t^{k-1}, U_t^k) \quad (1)$$

where $p(\mathbf{x}_k | \mathbf{x}_{k-1}, Z_o^{k-1}, U_o^k, Z_t^{k-1}, U_t^k)$ is obtained from the posterior of the previous step and the motions given by \mathbf{u}_o^k and \mathbf{u}_t^k ,

$$p(\mathbf{x}_k | \mathbf{x}_{k-1}, Z_o^{k-1}, U_o^k, Z_t^{k-1}, U_t^k) = \int p(\mathbf{x}_k | \mathbf{x}_{k-1}, \mathbf{u}_o^k, \mathbf{u}_t^k) p(\mathbf{x}_{k-1} | Z_o^{k-1}, U_o^{k-1}, Z_t^{k-1}, U_t^{k-1}) d\mathbf{x}_{k-1}$$

The previous formulation assumes independent Markov measurement processes for each platform and a Markov process for the evolution of \mathbf{x} . Equation (1) estimates the posterior at time k using the posterior computed at time $k-1$, a motion model $p(\mathbf{x}_k | \mathbf{x}_{k-1}, \mathbf{u}_o^k, \mathbf{u}_t^k)$ and the measurement models $p(z_o^k | \mathbf{x}_k)$ and $p(z_t^k | \mathbf{x}_k)$.

V. IMPLEMENTATIONS OF THE BAYES ESTIMATOR

In this Section we provide three different implementations for Equation (1): the classical Extended Kalman Filter (EKF), a sampled based approach known as particle filters (PF) and a combination of both techniques (PF-EKF). The EKF assumes Gaussian distributions and noises and requires to linearize the process and measurement models. The lack of an accurate initial relative location together with the use of bearing-only measurements makes the EKF unappropriated for the first stages of the estimation process.

On the other hand particle filters do not require any of the previous assumptions and are able to represent any kind of distribution given a sufficient number of particles. However, they are computationally more expensive and introduce a discretization error which depends on the number of particles used. Therefore, we propose an hybrid approach (PF-EKF) that uses a particle filter in the first steps and switches to an EKF when the distribution of the particles is close to a Gaussian.

The movement of the target platform seen from the observer is a combination of the movements of each platform $\mathbf{u}_t = (dx_t, dy_t, d\theta_t)$ and $\mathbf{u}_o = (dx_o, dy_o, d\theta_o)$ (see Figure 1(a)),

$$\mathbf{x}_k = f(\mathbf{x}_{k-1}, \mathbf{u}_o^k, \mathbf{u}_t^k) = \ominus \mathbf{u}_o^k \oplus \mathbf{x}_{k-1} \oplus \mathbf{u}_t^k \quad (2)$$

Note that \mathbf{u}_t^k and \mathbf{u}_o^k are noisy measurements of the true displacement of the robot corrupted with noises \mathbf{v}_t^k and \mathbf{v}_o^k respectively. The operators \oplus and \ominus represent the composition and inversion of locations,

$$\mathbf{x}_1 \oplus \mathbf{x}_2 = \begin{pmatrix} \cos \theta_1 x_2 - \sin \theta_1 y_2 + x_1 \\ \sin \theta_1 x_2 + \cos \theta_1 y_2 + y_1 \\ \theta_1 + \theta_2 \end{pmatrix} \quad (3)$$

$$\ominus \mathbf{x}_1 = \begin{pmatrix} -\cos \theta_1 x_1 - \sin \theta_1 y_1 \\ \sin \theta_1 x_1 - \cos \theta_1 y_1 \\ -\theta_1 \end{pmatrix} \quad (4)$$

On the other hand the observation models for each robot are,

$$z_o^k = h_o(\mathbf{x}_k) + w_o^k = \tan^{-1} \frac{y(k)}{x(k)} + w_o^k \quad (5)$$

$$z_t^k = h_t(\mathbf{x}_k) + w_t^k = \pi + \tan^{-1} \frac{y(k)}{x(k)} - \theta(k) + w_t^k$$

where w_o^k and w_t^k are the measurement noises. The measurement models illustrate the difference between obtaining observations from both platforms or just from a single one. In the latter case there is no information about the relative localization of the platforms. The filters presented below incorporate the observations provided by the platforms as they become available given that the motion of the platforms is known.

A. Extended Kalman filter implementation

In this Section we implement Equation (1) using an EKF. The EKF framework models \mathbf{x} with a Gaussian distribution and approximates the nonlinear process and

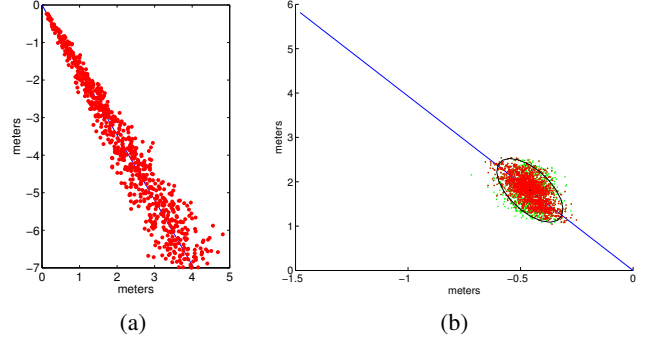


Fig. 2. (a) particles initial distribution according to the measurement model, (b) the distribution after some steps is close to a Gaussian

measurement equations using the partial derivatives of the nonlinear functions. We denote $\hat{\mathbf{x}}_{k|k}$ to the current estimate of the state vector \mathbf{x} at time k and $\mathbf{P}_{k|k}$ to the associated covariance matrix. The odometry readings, \mathbf{u}_t^k and \mathbf{u}_o^k , and the bearing measurements, z_o^k and z_t^k , are corrupted by independent zero-mean white Gaussian noises with covariance matrices \mathbf{V}_{t_k} , \mathbf{V}_{o_k} , \mathbf{R}_{o_k} and \mathbf{R}_{t_k} respectively. We use a first order approximation of the process and measurement models (Equations (2) and (5)):

$$\begin{aligned} f(\mathbf{x}, \mathbf{u}_o, \mathbf{u}_t) &\approx f(\hat{\mathbf{x}}, \mathbf{u}_o, \mathbf{u}_t) + \mathbf{F}_x(\mathbf{x} - \hat{\mathbf{x}}) + \mathbf{F}_{u_o} \mathbf{v}_o + \mathbf{F}_{u_t} \mathbf{v}_t \\ h_i(\mathbf{x}) &\approx h_i(\hat{\mathbf{x}}) + \mathbf{H}_i(\mathbf{x} - \hat{\mathbf{x}}) + w_i, \quad i \in [o, t] \end{aligned}$$

where \mathbf{F}_x , \mathbf{F}_{u_o} and \mathbf{F}_{u_t} are the Jacobians of the function f with respect to \mathbf{x} , \mathbf{u}_o and \mathbf{u}_t and \mathbf{H}_i is the Jacobian of the function h_i ($i \in \{o, t\}$) with respect to the state vector \mathbf{x} evaluated at the current state $\hat{\mathbf{x}}_k$ and motions \mathbf{u}_o^k and \mathbf{u}_t^k .

Using the previous Jacobians we provide next the resulting Kalman filter equations for time k . The predicted relative location $\hat{\mathbf{x}}_{k|k-1}$ and its associated covariance matrix $\mathbf{P}_{k|k-1}$ using the linearized process model are,

$$\begin{aligned} \hat{\mathbf{x}}_{k|k-1} &= f(\hat{\mathbf{x}}_{k-1|k-1}, \mathbf{u}_o^k, \mathbf{u}_t^k) \\ \mathbf{P}_{k|k-1} &= \mathbf{F}_x \mathbf{P}_{k-1|k-1} \mathbf{F}_x^T + \mathbf{F}_{u_o} \mathbf{V}_{o_k} \mathbf{F}_{u_o}^T + \mathbf{F}_{u_t} \mathbf{V}_{t_k} \mathbf{F}_{u_t}^T \end{aligned}$$

with \mathbf{F}_x , \mathbf{F}_{u_o} and \mathbf{F}_{u_t} evaluated at $\hat{\mathbf{x}}_{k-1}$, \mathbf{u}_o^{k-1} and \mathbf{u}_t^{k-1} . On the other hand the update step is preformed sequentially for each available observation,

$$\begin{aligned} \mathbf{S}_k &= \mathbf{H}_i \mathbf{P}_{k|k-1} \mathbf{H}_i^T + \mathbf{R}, & \mathbf{W}_k &= \mathbf{P}_{k|k-1} \mathbf{H}_i^T \mathbf{S}_k^{-1} \\ \hat{\mathbf{x}}_{k|k} &= \hat{\mathbf{x}}_{k|k-1} + \mathbf{W}_k v_k, & \mathbf{P}_{k|k} &= \mathbf{P}_{k|k-1} - \mathbf{W}_k \mathbf{S}_k \mathbf{W}_k^T \end{aligned}$$

where the innovation is $v_k = z_i^k - h_i(\hat{\mathbf{x}}_{k|k-1})$ and the Jacobian \mathbf{H}_i is evaluated at the current estimate of \mathbf{x} using the appropriated function h_i . When both observations are available, the $\hat{\mathbf{x}}_{k|k}$ and $\mathbf{P}_{k|k}$ obtained from the first update step are used to perform the second one.

B. Particle filter implementation

Particle filters are sequential Monte-Carlo techniques to estimate posterior distributions [5]. They represent the distributions by a set of M samples $S = \{\mathbf{x}_k^{[1]}, \dots, \mathbf{x}_k^{[M]}\}$. The usual way to implement the recursive Bayes estimator of Equation (1) is to use the posterior obtained in the previous step $k-1$ and the motion model to guess the distribution at time k . For each sample $\mathbf{x}_{k-1}^{[i]}$ a new sample



Fig. 3. (a) Images acquired by the robots. (b) Labmate and Scout robots.

$\mathbf{x}_k^{[i]}$ is generated from the possible locations described by \mathbf{u}_o^k and \mathbf{u}_t^k ,

$$\text{sample } \mathbf{x}_k^{[i]} \text{ from } p(\mathbf{x}_k | \mathbf{x}_{k-1}^{[i]}, \mathbf{u}_o^k, \mathbf{u}_t^k) \quad (6)$$

The set of all these samples conforms what is known as the proposal distribution. For each robot we use a motion model similar to the one described in the Carnegie Mellon Navigation toolkit [11]. Note that we are composing two uncertain motions. The number of particles must be sufficient to sample all the possible locations induced by both motions requiring more particles than a single one. The samples from the proposal are distributed according to $p(\mathbf{x}_k | Z_o^{k-1}, U_o^k, Z_t^{k-1}, U_t^k)$ and do not include the information of the last observations z_o^k and z_t^k . To take into account the difference between the proposal distribution and the target distribution $p(\mathbf{x}_k | Z_o^k, U_o^k, Z_t^k, U_t^k)$ the samples are weighed according to their likelihood,

$$\alpha^{[i]} = \frac{\text{target}}{\text{proposal}} \propto p(z_o^k | \mathbf{x}_k^{[i]}) p(z_t^k | \mathbf{x}_k^{[i]}) \quad (7)$$

The new set of particles approximating $p(\mathbf{x}_k | Z_o^k, U_o^k, Z_t^k, U_t^k)$ is then generated by sampling from the proposal distribution according to the importance factors $\alpha^{[i]}$.

The measurement model is the likelihood of the observation given the current state. We model the noise of each bearing as an independent zero mean Gaussian distribution with a standard deviation σ_w ,

$$\begin{aligned} p(z_o^k | \mathbf{x}_k^{[i]}) &= N(z_o; h_o(\mathbf{x}_k^{[i]}), \sigma_{w_o}) \\ p(z_t^k | \mathbf{x}_k^{[i]}) &= N(z_t; h_t(\mathbf{x}_k^{[i]}), \sigma_{w_t}) \end{aligned}$$

C. Combining both approaches

In order to benefit from the advantages of each implementation we propose to use a combination of both techniques. As we do not have any prior information about the relative location of the platforms, we use the measurements to initialize the filters. According to the measurement Equations (5) we do not have any information about the distance between the platforms. Information about their relative orientation is only available when both robots observe each other. During the first steps of the estimation

process a Gaussian distribution does not approximate well the uncertainty of the relative pose. Therefore we use a particle filter to sample all the possible initial locations (Figure 2(a)).

After a certain variable number of steps the particles converge to a unique mode corresponding to the truth relative pose. We periodically try to fit a Gaussian to the current particles of the filter. When the particles are close to a Gaussian distribution we switch to an EKF to track the relative pose of the robots (Figure 2(b)).

The complexity of the EKF is $\mathcal{O}(m^2)$ where $m = \dim(\mathbf{x})$. On the other hand the complexity of the PF is $\mathcal{O}(N)$ with N the number of particles. As m remains constant and is much smaller than the required number of particles N , the EKF implementation is computationally much more efficient for our problem. This strategy combines the more powerful representation capabilities of the PF needed in the initial steps of the algorithm due to the use of bearing-only measurements with the computational efficiency of the EKF when the particles concentrate around the true location.

VI. EXPERIMENTS

To evaluate the method we have implemented it on two robots equipped with omnidirectional cameras. We have considered two different configurations: only one robot has detection capabilities (heterogeneous robots, hardware failures, use of non omnidirectional cameras) (**1-OBS**), and both robots have detection capabilities (**2-OBS**). We also conducted an extensive test of the algorithm using simulated data to analyze the performances of the EKF, the PF and the PF-EKF in terms of robustness and precision for each configuration. In this Section we first present our experimental setup, then the results obtained with simulated data and finally those obtained with real data.

A. Setup

The robots, a TRC Labmate from HelpMate Robotics Inc and a Nomadic's Scout, are equipped with omnidirectional cameras based on a spherical mirror and on a hyperbolic mirror, respectively (see Figure 3(b)).

The images acquired by the omnidirectional cameras are dewarped to 360° -wide panoramic views. The dewarping is just a Cartesian to polar transformation when the camera and the mirror are vertically aligned [7], [6]. Both cameras

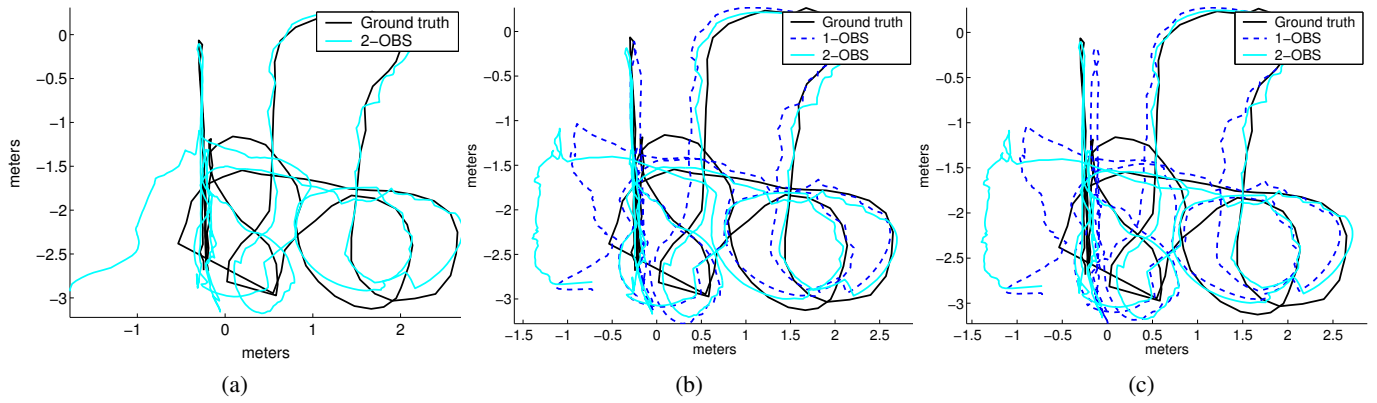


Fig. 4. Estimated trajectories using observations from both robots. (a) EKF implementation, (b) PF implementation (c) PF-EKF implementation.

produce 768×576 [pix] omnidirectional images, which are dewarped to 1091×148 [pix] (labmate) and 1014×197 [pix] (scout) panoramic views (see Figure 3(a)).

The panoramic views allow each robot to track the other platform and measure its azimuthal location (bearing). This process is simplified as the robots carry a light that can be easily detected in a region of interest. At the same time the robots acquire an image, the odometry is also observed. The data acquisition rate is 2Hz.

The ground truth of the trajectories of the robots is obtained using an external camera placed in a corner of the room close to the ceiling. The perspective view of the room is transformed to an orthographic view of the ground plane using an homography. The robots are tracked by hand, marking in the image of the ground plane their locations. The heading of the labmate is obtained from the orientation of its square-shaped footprint.

B. Simulations

We first tested our algorithm using simulated data. The main objective of these tests was to evaluate the accuracy and robustness of the EKF, the PF and the PF-EKF implementations. We generate 200 datasets where the robots moved at random speeds inside a 10×10 [m] squared room. Both the bearing measurements and the odometry readings were corrupted with Gaussian noise. We limited the detection of the other robot to a maximum range of 8 meters and generate observations at 2Hz to be consistent with our platform capabilities. The PF and PF-EKF implementations used 1000 particles.

The KF implementation is very sensitive in the first steps due to the big uncertainty in the initial relative location when only one observation is available (**1-OBS**). It does not always converge to the correct solution (only 60% of the simulated runs succeeded). In the **2-OBS** case, the EKF converged to the right solution in all the simulations due to the extra orientation information provided by the measurements of the target robot. On the other hand, the PF and PF-EKF implementations converged to the solution for all the datasets in both cases. Table I summarizes the results for the EKF, the PF and PF-EKF.

When they converged all the implementations estimate the relative position with similar accuracy. The differences

		μ_x [m]	σ_x [m]	μ_y [m]	σ_y [m]	μ_θ [rad]	σ_θ [rad]
1-OBS	PF	0.11	0.11	0.10	0.16	0.07	0.08
	PF-EKF	0.10	0.15	0.11	0.18	0.05	0.06
2-OBS	EKF	0.09	0.11	0.08	0.09	0.01	0.01
	PF-EKF	0.10	0.12	0.11	0.14	0.02	0.02

TABLE I

MEAN AND STD OF THE ERRORS USING SIMULATED DATA

obtained are not statistically significant. The mean execution times per iteration are 109 milliseconds for a particle filter step and around 0.1 milliseconds for a Kalman filter one. These times correspond to a Matlab implementation on a Pentium IV at 1.2 Ghz and illustrate the different complexities of each implementation.

The results suggest that the particle filter is necessary only in the first steps of the algorithm to cope with the non Gaussian uncertainty that arises from the first observations. Once the filter has converged around the true location, the EKF is able to track the position with less computational cost. The number of steps needed to switch to the EKF depends on the relative motion of the robots. Those trajectories that induce changes on the relative bearing are best suited for the localization task while those without changes provide less information. Furthermore, one can rely on the EKF implementation when both platforms provide observations of the other robot.

C. Real data

We also tested our method with real data obtained using the setup presented above. In the experiments the vehicles were driven manually during approximately 400 seconds.

The algorithm computes relative locations between the moving platforms. To plot the results we used the ground truth trajectory of the Labmate and the relative location estimation at each point in time to compute the trajectory of the Scout. Then we compare this trajectory with the Scout ground truth trajectory. The Labmate was the observer platform and the Scout the target one.

Figures 4(b) and (c) show the results of the PF and the PF-EKF implementations for the **1-OBS** and the **2-OBS** cases. As in the simulated runs the results obtained for each configuration are slightly better for the **2-OBS** case. The results of the EKF implementation in the **2-OBS**

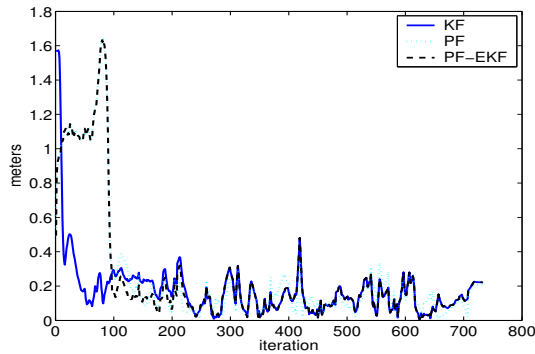


Fig. 5. Error evolution using observations from both robots

configuration are shown in Figure 4(a). Table II shows the mean and the standard deviation of the error between the estimated position and the ground truth after convergence (no orientation information is available from the ground truth due to the Scout symmetries).

		$\mu_x [m]$	$\sigma_x [m]$	$\mu_y [m]$	$\sigma_y [m]$
1-OBS	PF	0.08	0.06	0.12	0.09
	PF-EKF	0.11	0.07	0.15	0.10
2-OBS	EKF	0.07	0.05	0.12	0.09
	PF-EKF	0.07	0.06	0.11	0.09

TABLE II

MEAN AND STD OF THE ERRORS USING REAL DATA

Figure 5 shows the evolution of the position error for each implementation in the **2-OBS** case. Once the method has converged the maximum error is around 0.4 meters for all the implementations. The error peaks correspond to those parts of the trajectory where the robots are further away (up to five meters). The further the robot is, the less informative is the observation in terms of possible $x - y$ positions. This increased the estimation error and the uncertainty of the relative location when the platforms moved away one from the other.

The results show that the PF implementation is more robust than the EKF to estimate the initial relative location. This is specially important when only one observation is available. Once the filter converges to the true relative location, all the methods provide similar results. The PF-EKF implementation provides the best compromise between robustness and efficiency.

VII. CONCLUSIONS

We have presented a method to relatively localize pairs of robots fusing bearing measurements and the motion of the vehicles. Bearings are obtained as direct readouts of the omnidirectional cameras. This is convenient as compared to measuring depth, which would require knowledge or reconstruction of the world / robot structure. The method process the bearing measurements as they become available from each robot. A single camera is enough to estimate the relative pose allowing heterogeneous teams and increasing the robustness against failures.

We have proposed three different implementations of the recursive Bayes filter based on an Extended Kalman filter, a particle filter and a combination of both techniques. The

latter combines the benefits of each type of filter resulting in a robust and fast algorithm. We have shown several experimental results validating the solution to the problem and evaluating the different implementations.

As future work we plan to study the observability of the system and the generation of motions that improve the accuracy of the relative localization. We are also integrating a general tracker to acquire the bearings using pattern recognition techniques.

VIII. ACKNOWLEDGMENTS

This work has been partially supported by the Spanish projects MCYT-DPI2003-7986 and HP2002-0037, and by the Portuguese project Omnisys POSI/SRI/41506/2001 and the FCT POSI program in the frame of QCA III.

REFERENCES

- [1] Y. Bar-Shalom, XR Li, and T. Kirubarajan. *Estimation with Applications to Tracking and Navigation*. J. Wiley and Sons, 2001.
- [2] A. Blake and M. Isard, editors. *Active Contours: The Application of Techniques from Graphics, Vision, Control Theory and Statistics to Visual Tracking of Shapes in Motion*. Springer-Verlag, 1998.
- [3] A. Davison. Real-time simultaneous localisation and mapping with a single camera. In *IEEE Int. Conf. on Computer Vision*, pages 1403 – 1410 vol.2, 2003.
- [4] A.J. Davison and D.W. Murray. Mobile robot localisation using active vision. In *European Conf. on Computer Vision*, 1998.
- [5] A. Doucet, N. Freitas, and N.J. Gordon. *Sequential Monte Carlo Methods In Practice*. Springer Verlag, 2001.
- [6] J. Gaspar, N. Winters, and J. Santos-Victor. Vision-based navigation and environmental representations with an omni-directional camera. *IEEE Trans. on Robotics and Automation*, 16(6):890–898, 2000.
- [7] C. Geyer and K. Daniilidis. A unifying theory for central panoramic systems and practical applications. In *European Conf. on Computer Vision*, pages 445–461, Ireland, 2000.
- [8] A. Howard, M.J. Mataric, and G. Sukhatme. Putting the ‘i’ in ‘team’: an ego-centric approach to cooperative localization. In *IEEE Int. Conf. on Robotics and Automation*, Taiwan, 2003.
- [9] D. Koller, K. Daniilidis, and H.-H. Nagel. Model-based object tracking in monocular image sequences of road traffic scenes. *Int. Journal of Computer Vision*, 10(3):257–281, June 1993.
- [10] D. Lowe. Robust model-based motion tracking through the integration of search and estimation. *Int. Journal of Computer Vision*, 8(2):113–122, August 1992.
- [11] M. Montemerlo, N. Roy, and S. Thrun. Perspectives on standardization in mobile robot programming: The carnegie mellon navigation (CARMEN) toolkit. In *Proc. of the Conf. on Intelligent Robots and Systems*, 2003.
- [12] L. Montesano, L. Montano, and W. Burgard. Relative localization for pairs of robots based on unidentifiable moving landmarks. In *Int. Conf. on Intelligent Robots and Systems*, Sendai, Japan, 2004.
- [13] H. Murase and S. K. Nayar. Visual learning and recognition of 3D objects from appearance. *Int. J. Computer Vision*, 14(1):5–24, January 1995.
- [14] C. Musso, N. Oudjane, and F. Le Gland. *Sequential Monte Carlo Methods In Practice*, chapter Improving Regularised Particle Filters. Springer, 2001.
- [15] K. Okuma, A. Taleghani, N. de Freitas, J. Little, and D. Lowe. A boosted particle filter: Multitarget detection and tracking. In *European Conf. on Computer Vision*, pages 28–39 vol.1, 2004.
- [16] P. Pinheiro and P. Lima. Bayesian sensor fusion for cooperative object localization and world modeling. In *Conf. on Intelligent Autonomous Systems*, Amsterdam, The Netherlands, 2004.
- [17] S. Roumeliotis and G.A. Bekey. Distributed multirobot localization. *IEEE Trans. on Robotics and Automation*, 18(5), 2002.
- [18] Stephen Se, D. Lowe, and J. Little. Mobile robot localization and mapping with uncertainty using scale-invariant visual landmarks. *Int. Journal of Robotics Research*, 21(8):735–758, 2002.
- [19] J. Spletzer, K. Das, R. Fiero, C.J. Taylor, V. Kumar, and J.P. Ostrowski. Cooperative localization and control for multi-robot manipulation. In *Proc. of the Conf. on Intelligent Robots and Systems*, Hawaii, USA, 2001.