Auton Robot (2007) 23: 305–324
DOI 10.1007/s10514-007-9049-4

Guidance of a mobile robot using an array of static cameras located in the environment

Ignacio Fernández · Manuel Mazo · José L. Lázaro · Daniel Pizarro · Enrique Santiso · Pedro Martín · Cristina Losada

Received: 26 May 2006 / Accepted: 23 July 2007 / Published online: 8 September 2007
© Springer Science+Business Media, LLC 2007

Abstract This paper presents a new proposal for positioning and guiding mobile robots in indoor environments. The proposal is based on the information provided by static cameras located in the movement environment. This proposal falls within the scope of what are known as intelligent environments; in this case, the environment is provided with cameras that, once calibrated, allow the position of the robots to be obtained. Based on this information, control orders for the robots can be generated using a radio frequency link. In order to facilitate identification of the robots, even under extremely adverse ambient lighting conditions, a beacon consisting of four circular elements constructed from infrared diodes is mounted on board the robots. In order to identify the beacon, an edge detection process is carried out. This is followed by a process that, based on the algebraic distance, obtains the estimated ellipses associated with each element of the beacon. Once the beacon has been identified, the coordinates of the centroids for the elements that make up the beacon are obtained on the various image planes. Based on these coordinates, an algorithm is proposed that takes into account the standard deviation of the error produced in the various cameras in ascertaining the coordinates of the beacon's elements. An odometric system is also used in guidance that, in conjunction with a Kalman Filter, allows the position of the robot to be estimated during the time intervals required to process the visual information provided by the cameras.

Keywords Array of cameras · Intelligent environments · Mobile robot guidance · Infrared beacons

I. Fernández (✉) · M. Mazo · J.L. Lázaro · D. Pizarro · E. Santiso · P. Martín · C. Losada
Electronics Department, University of Alcalá, Campus Universitario s/n, Alcalá de Henares, 28805 Madrid, Spain
e-mail: ifdez@depeca.uah.es

M. Mazo
e-mail: mazo@depeca.uah.es

J.L. Lázaro
e-mail: lazaro@depeca.uah.es

D. Pizarro
e-mail: pizarro@depeca.uah.es

E. Santiso
e-mail: santiso@depeca.uah.es

P. Martín
e-mail: martin@depeca.uah.es

C. Losada
e-mail: losada@depeca.uah.es

1 Introduction

In the field of autonomous robot guidance, one of the tasks that is currently attracting greatest attention is that of ascertaining the robot's position and orientation within the environment with sufficient accuracy. Several methods have been developed (Borestein et al. 1996; Santiso 2003) to carry out this task.

The various localization methods may be classified into two broad groups: those that require sensors to be incorporated within the work environment and those that incorporate sensors on board the robot. The first alternative offers significant advantages given that, in many cases, it allows the complexity of the robots to be notably reduced; it facilitates simultaneous navigation by multiple robots within the same environment; and it provides information on what is happening within the environment and so allows decisions to be made according to this information. This alternative includes "intelligent environments" (Lee et al. 2001;

Steinhaus et al. 2004), which are characterized by the use of an array of sensors located in fixed positions throughout the environment and that are distributed strategically to cover the robots' entire field of movement. The information provided by the sensors should allow both the robots' position and that of any other object found within the environment to be ascertained. Numerous papers (Borestein et al. 1996; Santiso 2003; Villadangos et al. 2004; Ocaña et al. 2005; López et al. 2005) have been written on the alternative in which the sensors are mounted on board the robots. In general, all of these solutions require the robots to carry extremely sophisticated onboard systems with which to process the information that the sensors capture from the environment. Furthermore, the most significant disadvantage that this alternative presents is that it requires explicit models of the environment.

The intelligent environment approach to mobile robot guidance offers a series of major advantages in applications with indoor environments, among which the following may be highlighted: the information obtained from the robot's movement environment is more complete; the robots' onboard systems are notably simplified; it only requires a single infrastructure to be incorporated within the environment (independently of the number of robots in operation within the environment); the costs are notably reduced, above all, when several robots are operating within the environment; and it is possible to obtain information about the position of all of the robots, thereby facilitating cooperation between them.

Despite the fact that 3D object tracking is by itself a classical problem in computer vision, few papers have been published on mobile robot guidance using an external camera ring. The most significant approaches can be divided into two separated groups; the criterion is based on the previous knowledge that is required about the robot.

The first group includes those works that are capable of using the robot's natural appearance and camera geometry to obtain the position. Such methods don't rely on previous knowledge or invasive landmarks to accomplish the task. In (Sogo et al. 1999) and (Kruse and Wahl 1998), robot appearance is previously learnt for tracking. In (Hoover and Olsen 2000) and (Steinhaus et al. 1999), the position of static and dynamic obstacles is obtained by multiple camera fusion inside an occupancy grid. In (Pizarro et al. 2005) have used a set of fiducials to recover metric pose and structure simultaneously. Most of this novel approaches unless promising, still fail in terms of stability, complexity or robustness required in industrial applications.

The second group makes use of strong prior knowledge by using artificial landmarks attached to the robot. In (Morioka et al. 2006; Lilienthal and Duckett 2003) and (Yun et al. 2004), a set of color landmarks are used for robot localization. In (Hoff et al. 1996) and (Naimark and Foxlin 2002), a contrast template is used for tracking. Both approaches are naturally affected by lighting variations. To overcome sensitivity to light changes, in (Chung et al. 2001) a reflectance material is used as an artificial landmark. Takase et al. (2003), Hada et al. (2002) and Terado and Fujiwara (2004) use active infrared beacons in robot localization and tracking applications. Such methods offer a certain degree of robustness and the matching process is dramatically simplified. Despite that, important issues as marker subpixel detection or multiple camera uncertainty fusion are not deeply developed in such works. In fact, existing commercial capture motion systems as (ViconPeak Online; AICON 3D Online) or (Metronor Online), also based on IR markers, achieve better performance in terms of accuracy and marker identification. The main drawbacks of commercial systems derive from its application, which consist of detecting human non rigid geometry. Such requirements usually overload the complexity in the system in terms of the number of markers and the need for modulation codes for marker identification. The result directly affects both the overall cost of the sensors and hardware.

The proposal presented in this work obtains robot position and orientation $(X, Y, Z, \theta_X, \theta_Y, \theta_Z)$ by using only calibrated cameras placed within the environment and a simple beacon onboard the robot (built by using four infrared diodes arranged in "T" shape). The method achieves the pose robustly even against changes in the lighting conditions and by consuming a reduced processing time. A robot's beacon detection and matching is solved by using a robust segmentation approach which consists of edge detection and ellipse fitting (Image projection model of each infrared element of the beacon). In order to obtain a robot pose from images retrieved from multiple calibrated cameras, a robust cost function is proposed which makes use of the uncertainty present in the information taken from the set of cameras. Optimization of the proposed cost function allows an optimal pose computation. Information from the cameras is weighted so that the more the uncertainty, the less the influence in the pose solution.

This proposal is included inside those approaches which make use of strong prior knowledge by using artificial landmarks attached to the robot. By using infrared elements to build a beacon it is possible to achieve a high immunity to illumination changes and to the presence of light spots in the environment in which the robot moves.

Comparing to those referents that use the same approach, (Takase et al. 2003; Hada et al. 2002) and (Terado and Fujiwara 2004), the proposed solution is able to fuse information between the different cameras taking into account the inherent uncertainty in detection. In addition, the projection model of the infrared beacon allows reaching subpixel accuracy in image detection, which is translated into better metric accuracy in a robot's pose compared to previous methods

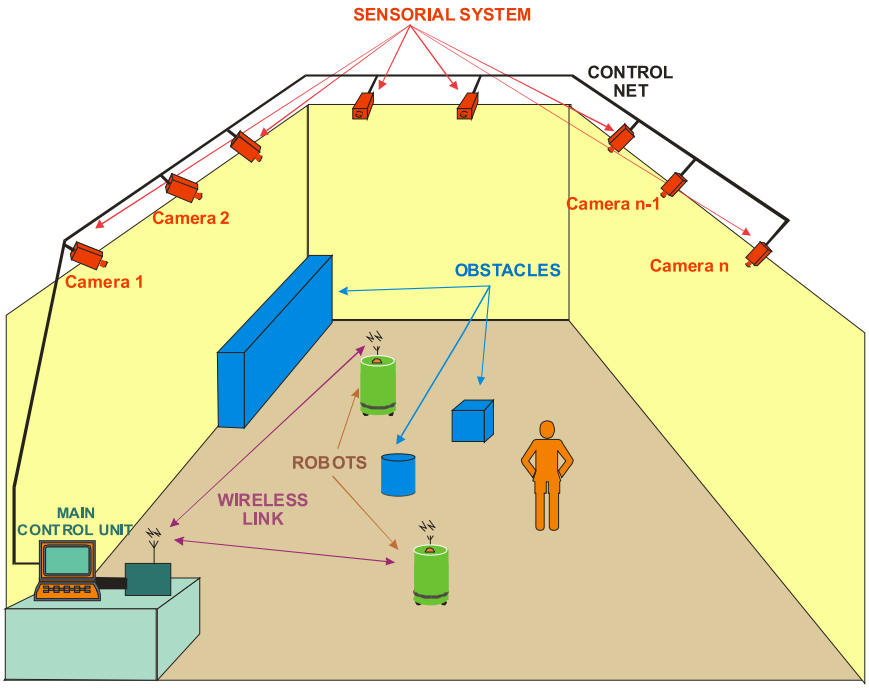


Fig. 1 General configuration of the guidance system using cameras located within the work environment

with similar approaches. Compared to commercial tracking and motion capture systems, the proposed method still keeps versatility and simplicity in the design. It thus allows avoidance of complex processing techniques for landmark detection but it still achieves high accuracy in a robot's pose even in the case that the robot to camera distance is large. As a conclusion, the proposal presented in this paper allows obtaining a robot's pose in a robust and confident fashion, achieving high accuracy with low complex algorithms.

As another remark, the proposed method includes a calibration procedure to obtain extrinsic and intrinsic camera parameters with high accuracy. Finally, a set of satisfactory experiments are taken inside real environments as a proof of concept. The system thus can be easily applied to different applications as industrial robots, aids to handicapped people, surgery robots..., with a reduced cost.

This paper consists of seven sections, including the introduction. Section two provides a general description of the solution that has been developed, as well as a description of the onboard beacon and a description of the camera model used. Section three describes the algorithm used to detect the beacon mounted on board the robot. Section four presents the algorithm used to ascertain the position of the robot. Section five describes the guidance system. Section six presents the results obtained, and finally, section seven presents the conclusions.

2 General Description of the Proposed System

The general configuration of the proposed solution is shown in Fig. 1. The key elements of the same consist of a vision system (cameras) responsible for capturing the environment, and a main control unit responsible for processing the images provided by the vision system, obtaining the robots' position and generating the control set-points that are sent to the robot by radio. Evidently, the most important aspects related to the proposed guidance solution are those related to the segmentation of the beacon on board the robot and to obtaining the robot's 3D coordinates and orientation within the movement environment. Both aspects are closely linked to the type of beacon and cameras used, as well as to the calibration of the latter.

This section describes the beacon to be mounted on board the robot, the camera model employed and the process used to calibrate a multi-camera system.

2.1 Infrared (IR) beacons

In order to detect and identify the robots using external cameras, different alternatives have traditionally been used based on beacons mounted on the robots. The type and shape of these beacons may vary widely and range from colored barcodes (Lee et al. 2001) and black and white circles (McNamee 2003) through to color circles (Huabin et al. 2004) and other shapes (Fiala 2004). All of the beacons proposed to date by other researchers are characterized by their simplicity, even though in some cases, accurate capture of the beacons is extremely sensitive to the environment's lighting

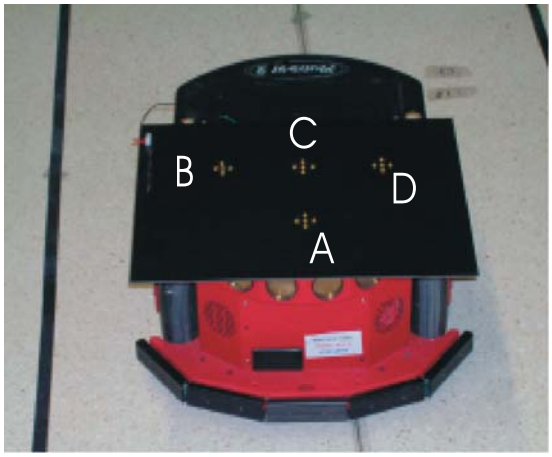


Fig. 2 Arrangement of the four elements (constructed from IR diodes) that constitute the beacon used to make an easy identification of the robots and to obtain their position and orientation

conditions. Although there is no reason why these conditions should be highly changeable in indoor environments, in some cases, there may nevertheless be changes in artificial lighting, or changes arising from natural lighting when the robot travels through environments with windows or with areas of greater brightness, etc. that may represent a significant difficulty as regards detection of the beacons. It should also be taken into account that identification of the beacon, and as a consequence obtaining its 3D coordinates, becomes more difficult as the distance between the robots and the cameras increases.

As a result, this paper proposes the use of a beacon consisting of four circular elements with a 2 cm diameter. It proposes arranging them in a "T" shape (Fig. 2) and constructing them from medium power infrared (IR) diodes (10 mW) with a wavelength of 850 nm (each element of the beacon is made up of 5 diodes). Using this approach, it has been possible to verify in the practical tests carried out that the beacon's four elements are easily detectable, even in environmental conditions in which there are significant changes in lighting, and without the need to employ complex image processing algorithms. The use of four elements arranged in a "T" shape to form the beacon allows the robots to be identified (by providing redundant information) and their orientation obtained without needing to obtain this information from the robot's movements.

In this proposal, information about the robots' position and orientation is obtained at all times from the coordinates of the centroids of each of the beacon's four elements within the image planes of the various cameras.

Figure 2 shows the physical arrangement of the four elements that make up the IR beacon. All of these elements are similar and are identified by the letters A, B, C and D. The centroid of element C corresponds to the robot's position whilst the orientation of the same corresponds to that of the vector that runs from the centroid of C to that of A. Elements B and D are required in order to be able to locate the other two (C and A).

This type of beacon facilitates accurate determination of the robot's position and orientation. However, as the distance between the robot and the cameras increases, the separation on the image plane between the four elements that make up the beacon diminishes, and these may even come to overlap. In our case, the maximum permissible distance between the cameras and the robot is 8 meters. Another possible alternative, also tested in this paper, is to use a single group of diodes (a beacon consisting of a single element), which only allows the position of the robots (not their orientation) to be detected. In this case, using the same dimensions for the element of the beacon (2 cm diameter) the distance may be extended to 16 meters.

2.2 Camera calibration

Due to the influence that it has on determining the robots' position, the calibration of the multiple camera system (intrinsic and extrinsic parameters) is an important aspect.

The model selected for each camera is that of a pinhole perspective projection and includes distortion of the optic, given that this distortion may have a significant impact on accurate capture of the coordinates of the centroids of the beacon's elements. Thus, for point $\boldsymbol{P}$ in the scenario the distortion introduced by the optic means that said point, which in the absence of distortion would be projected at point $\boldsymbol{p}$ at coordinates (x,y) on the image plane, is instead projected at point $\boldsymbol{p'}$ at coordinates (x',y'). The relationship between the coordinates of both points, according to the photogrammetry, is:

$$\begin{aligned} x' &= x + x(k_1 r^2 + k_2 r^4) + 2b_1 xy + b_2(r^2 + 2x^2), \\ y' &= y + y(k_1 r^2 + k_2 r^4) + b_1(r^2 + 2y^2) + 2b_2 xy \end{aligned} \tag{1}$$

where k_1 and k_2 are the coefficients that model the radial distortion, b_1 and b_2 model the tangential distortion and $r = (x^2 + y^2)^{1/2}$. Equation (1) may be expressed in matrix form as:

$$\boldsymbol{p'} = \boldsymbol{p} + \delta(\boldsymbol{p}) \tag{2}$$

where $\delta(\boldsymbol{p})$ reflects the distortion of point $\boldsymbol{p}$.

In order to carry out the distortion correction process (capture of coordinates (x,y) at point $\boldsymbol{p}$ based on coordinates (x',y')), several different alternatives exist (Heikkilä 2000). In this case, the proposal made by Melen (1994) has been used, as it has been demonstrated that it provides good results and has a low computational cost. The algorithm proposed by Melen is iterative, following the sequence indicated in (3), where $\delta(\boldsymbol{p}) = \boldsymbol{p'} - \boldsymbol{p}$ represents displacement of the pixel coordinates due to the distortion. The iterative

process comes to an end when $p_{t+i} - p_{t+i-1} \leq 0.05$ pixels. In practice, it has been verified that three iterations are sufficient to achieve a good approximation in the correction of the distortion.

$$\begin{aligned} \boldsymbol{p}_t &= \boldsymbol{p}' - \delta(\boldsymbol{p}), \\ \boldsymbol{p}_{t+1} &= \boldsymbol{p}' - \delta(\boldsymbol{p}' - \delta(\boldsymbol{p})), \\ \boldsymbol{p}_{t+2} &= \boldsymbol{p}' - \delta(\boldsymbol{p}' - \delta(\boldsymbol{p}' - \delta(\boldsymbol{p}))), \\ \boldsymbol{p}_{t+3} &= \boldsymbol{p}' - \delta(\boldsymbol{p}' - \delta(\boldsymbol{p}' - \delta(\boldsymbol{p}' - \delta(\boldsymbol{p})))), \\ \boldsymbol{p}_{t+i} &= \cdots . \end{aligned} \tag{3}$$

The equation that relates the 3D coordinates of a point in the scenario to the coordinates on the image plane, given by (4), is taken as the basis from which to obtain the parameters (intrinsic and extrinsic) of each of the cameras.

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \lambda \cdot \boldsymbol{R}^{-1} \cdot \begin{bmatrix} \frac{u-u_0}{f_x} \\ \frac{v-v_0}{f_y} \\ 1 \end{bmatrix} - \boldsymbol{R}^{-1} \cdot \boldsymbol{T} \tag{4}$$

where (X, Y, Z) are the 3D coordinates of a point in the scenario, u and v are the coordinates on the image plane, f_x, f_y, u_o, and v_o are the camera's intrinsic parameters, $\boldsymbol{T} = [T_X, T_Y, T_Z]^T$ represents the translation between the global reference system and that of the camera, $\boldsymbol{R}$ is a 3×3 orthogonal matrix defined by the Euler angles (α, β, γ) and λ is a scale factor. The parameters $T_X, T_Y, T_Z, \alpha, \beta$ and γ are the camera's extrinsic parameters (rotation-translation between the global reference system and the camera's reference system).

In our case, the calibration process has been carried out in two phases, by using a single calibration pattern. In the first phase, the intrinsic parameters are obtained with the cameras away from their definitive locations (the process is carried out in a laboratory calibration environment), thereby enabling these parameters to be obtained to a high degree of precision. In the second phase, the extrinsic parameters ($\boldsymbol{R}$ and $\boldsymbol{T}$) are obtained between each camera and the global reference system (Fernández et al. 2002). This latter process is carried out with the cameras installed in their definitive locations and using the intrinsic parameters obtained in the first phase. In order to obtain $\boldsymbol{R}$ and $\boldsymbol{T}$, the photogrametric calibration method developed in (McNamee 2003) has been used.

In order to obtain extrinsic parameters from the set of cameras, named as $\boldsymbol{R}_i, \boldsymbol{T}_i$ and where subindex i identifies the camera, a calibration pattern is placed in a known pose $(\boldsymbol{R}_{pw}, \boldsymbol{T}_{pw})$ with respect to a world reference frame. It is shown in Fig. 3 in the presence of $n = 3$ cameras. The po-

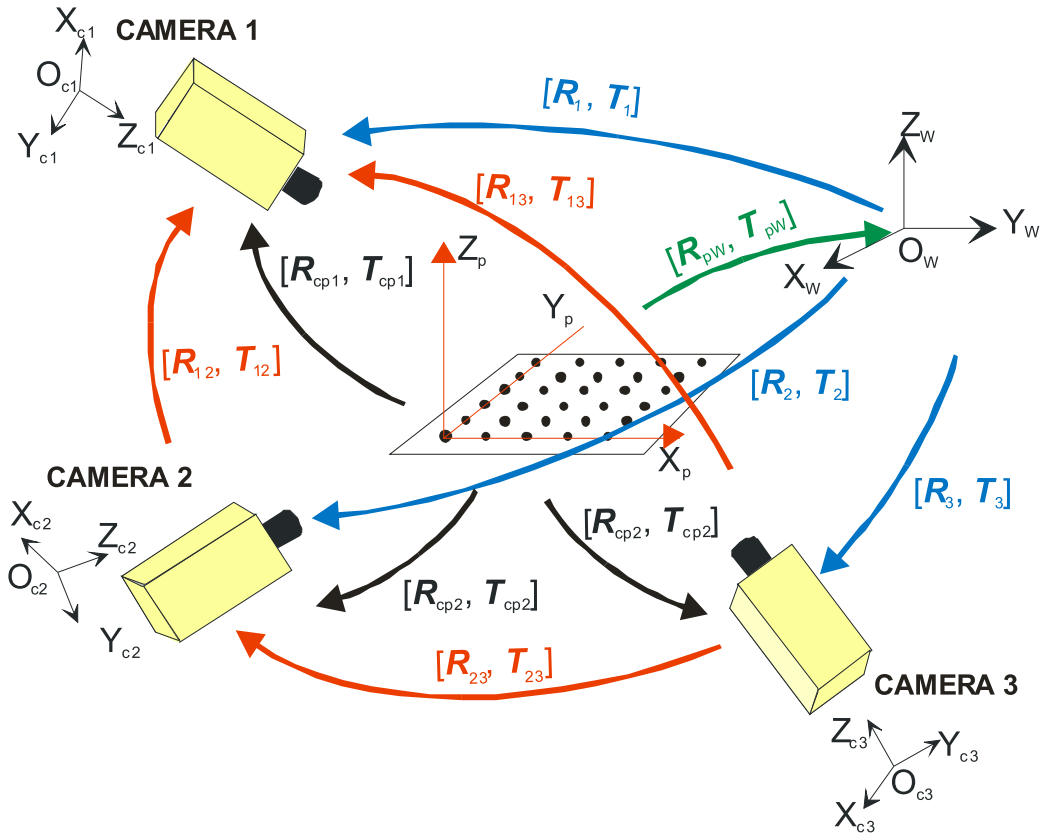


Fig. 3 Relationship between world coordinate origin, the calibration pattern and the cameras

sition of the calibration pattern must allow to be observed by a minimum number m of cameras ($m = 3$ in Fig. 3) from the total number n. The values of $\boldsymbol{R}_{pw}$, $\boldsymbol{T}_{pw}$ have been obtained by using a theodolite. From the analysis of the pattern points captured from the set of cameras, the values of $\boldsymbol{R}_{cpi}$ and $\boldsymbol{T}_{cpi}$ are obtained by using an algorithm similar to the one proposed in (McNamee 2003). By combining $\boldsymbol{R}_{cpi}$, $\boldsymbol{T}_{cpi}$ and $\boldsymbol{R}_{pw}$, $\boldsymbol{T}_{pw}$ as is shown in (5), the extrinsic parameters of each camera from the world coordinate origin ($\boldsymbol{R}_i$, $\boldsymbol{T}_i$) are obtained.

$$\begin{aligned} \boldsymbol{R}_i &= \boldsymbol{R}_{cpi} \cdot \boldsymbol{R}_{pg}^{-1}, \\ \boldsymbol{T}_i &= \boldsymbol{T}_{cpi} - \boldsymbol{T}_{pg} \cdot \boldsymbol{R}_{cpi} \cdot \boldsymbol{R}_{pg}^{-1}. \end{aligned} \tag{5}$$

Once $\boldsymbol{R}_i$, $\boldsymbol{T}_i$ are available for the set of m cameras, the rotation and translation parameters between each pair of cameras $\boldsymbol{R}_{ij}$ and $\boldsymbol{T}_{ij}$ ($i \neq j, i, j = 1, 2, \ldots, m$) are easily computed in (6). If one position of the pattern is not enough to obtain the relative positions of the cameras, the calibration pattern must be moved to a new position and the calibration procedure is repeated until the complete set of extrinsic parameters is known.

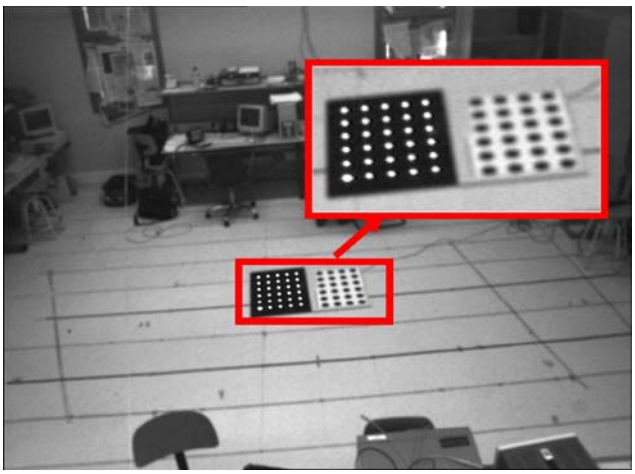
$$\boldsymbol{R}_{ij} = \boldsymbol{R}_i \cdot \boldsymbol{R}_j^{-1} \quad \text{and} \quad \boldsymbol{T}_{ij} = \boldsymbol{T}_i - \boldsymbol{R}_i \cdot \boldsymbol{R}_j^{-1} \cdot \boldsymbol{T}_j. \tag{6}$$

Calibration is not carried out in a single step because the calibration templates need to be captured by the various cameras and, in environments with large dimensions, this makes it necessary to locate them at significant distances from said cameras. As a consequence, the degree of precision achieved in the capture of the intrinsic parameters would not be sufficiently high.

As regards the calibration template, it is essential to be aware of the importance of both the shape of the elements that make up the template and the number of these (Lavest et al. 1998). In our case, taking into account the long distances between the cameras and the template, and bearing in mind that it is not possible to use large-sized templates, we have opted to use an active template constructed from IR diodes. This consists of luminous circles (5 mm diameter, consisting of a single infrared diode) on a black background. It is compared with a passive template consisting of black circles (6 centimeter diameter) on a white background. As may be observed, the quality as regards perception of the calibration points on the active template is greatly superior.

To retrieve the intrinsic parameters of each camera, a set of 10 images of the calibration pattern is obtained for each camera. In those images, it is of importance that the observed calibration pattern must take the maximum amount of the image area. In Fig. 4(a) an example image of the pattern is shown for calibrating intrinsic parameters. In Fig. 4(b) the image of the pattern is shown for the case of calibration of extrinsic parameters.

(a) Example of image used for intrinsic parameter calibration

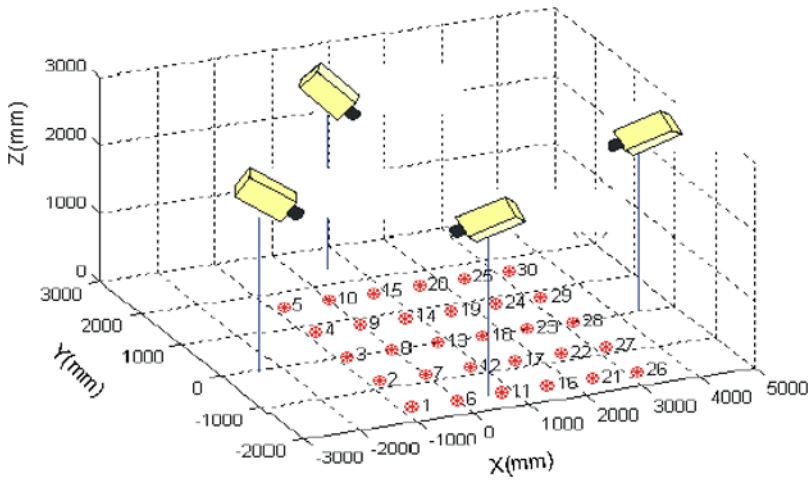


(b) Example of image used for extrinsic parameter calibration

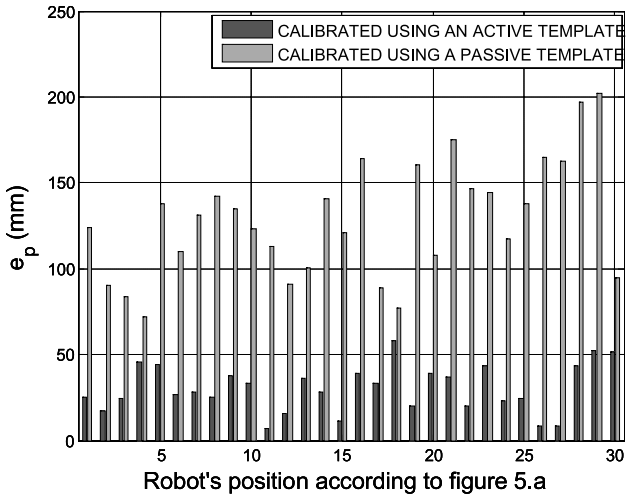
Fig. 4 Images used for calibrating the cameras

In Fig. 4(b), it is shown that, despite the distance between the pattern and the camera, the calibration pattern is clearly detected with enough contrast. The active pattern is thus essential for a successful calibration at long distances, where it doesn't take place over the whole image.

In order to verify the effectiveness of both the calibration method and the calibration template used, once calibration had been carried out, position measurements for the beacon on board the robot were taken for different positions of the same within the environment, as it's shown in Fig. 5(a), a set of 30 test poses (numbered from 1 to 30) where chosen in uniform distribution inside the motion environment of the robot. Figure 5(b) shows robot's position error (e_p) for each pose (1 to 30) in Fig. 5(a). The magnitude (e_p) of error position is defined as the Euclidean distance between the real position and the measured position. The ground truth for the position of the pattern was measured with a theodolite. Two tests have been carried out, one using an active template and another using a passive template.



(a) Test points localization



(b) Localization error in the robot as a function of the kind of pattern used in the calibration method. Dark bars show the result with the proposed active calibration pattern composed by IR diodes. Light color bars show the error with a passive pattern

Fig. 5 Measurement of the position error using the camera parameters obtained when using the two calibration templates

Novelty and accuracy of the proposed calibration system comes from the use of an active calibration pattern, which allows attainment of subpixel accuracy in the calibration points even at long distances. By dividing the process into two steps for obtaining first intrinsic and second extrinsic parameters, a better performance is achieved. In the first step, a high accuracy in the intrinsic parameters occurs by using close distance images of the pattern. In the second step the use of a theodolite for the true pattern position in the world frame, provides a method to obtain a good estimation of the extrinsic parameters.

3 Beacon Detection Algorithm

Using an IR beacon on board the robot helps greatly, regarding the image processing algorithms and drastically reduces processing time. In our case, the proposed algorithm used to locate the beacon includes four phases:

- Selection of a window of interest around the beacon according to the robot's predicted position (this eliminates the influence of noise from outside of the window and reduces calculation time).
- Segmentation of the beacon within the window of interest.
- Identification (classification) of the beacon within the segmented objects.
- Obtaining of the centroid of the four elements that make up the beacon.

Each of these phases is described below.

3.1 Selection of the window of interest

The window of interest is a square window defined on the image plane around the beacon. The position of the window at instant of time $k+1$ is based on the prediction made by a Kalman Filter at instant k. As regards the dimensions of the window, these are dynamic, so if the beacon is not detected within the initial window, the size of the window is increased until its maximum dimensions are reached (total size of the image). Logically, when the system is launched for the first time, the window of interest will cover the entire image. However, after this initial process, the size of the window will vary according to the predicted position of the beacon (robot) generated by the Kalman Filter. The minimum size of the window of interest is determined by the rectangle that contains the beacon.

3.2 Segmentation of the beacon within the window of interest

The segmentation process is initiated within the image captured by each camera by performing detection of the edges of the objects present in the window of interest. This solution is justified by the fact that the four elements that make up the IR beacon present a very high grey level compared with the rest of the image and, therefore, edge detection will further highlight these edges, which will facilitate later segmentation. A Prewitt-type 3×3 mask has been used for edge detection.

The segmentation process has been carried out by performing thresholding using an adaptive threshold. Thus, at every instant k the threshold value, T_k, is obtained from the histogram of the window of interest. In our case, we have used an adaptive threshold based on the Otsu method (Liao et al. 1999). Figure 6 shows the results of segmentation of the window of interest.

3.3 Identification (classification) of the beacon

Once the segmentation process has been completed, and as a step prior to obtaining the centroid of the four elements that make up the beacon, the characteristics that allows the beacon to be identified within the series of objects existing within the window of interest, which are obtained following segmentation, are then obtained.

In the first phase, any objects of reduced area are eliminated, taking into account the size of the elements that make up the beacon, the characteristics of the optic used and the distance from the cameras.

Given that the elements that make up the beacon are circular and their projections on the image plane will be elliptical, a scan for possible ellipses is carried out. For each of the objects segmented within the window of interest, the ellipse that best matches the same, and that we shall call the "estimated ellipse", is obtained. The possible ellipses are identified based on the algebraic distance from the edge points of each segmented object.

Fig. 6 Result of carrying out thresholding within an image's window of interest

As is known, the algebraic distance from point $\boldsymbol{p}$ to curve $\boldsymbol{g}(\boldsymbol{p}, \boldsymbol{w}) = 0$ is given by $|\boldsymbol{g}(\boldsymbol{p}, \boldsymbol{w})|$. It is thus a case of finding the ellipse characterized by parameter vector $\boldsymbol{w} = [a, b, c, d, e, f]^T$ that best matches an object with N edge points and pixel coordinates $\boldsymbol{p}_i = [u_i, v_i]$, $i = 1, 2, \ldots, N$.

As the algebraic distance is given by:

$$d_w = \min_{\boldsymbol{w}} \sum_{i=1}^{N} \left| \boldsymbol{x}_i^T \cdot \boldsymbol{w} \right| \tag{7}$$

and that, with the restriction $4ac - b^2 > 0$, the equation for the ellipse is given by (8),

$$\begin{aligned} \boldsymbol{g}(\boldsymbol{p}, \boldsymbol{w}) &= \boldsymbol{x}^T \cdot \boldsymbol{w} \\ &= au^2 + buv + cv^2 + du + ev + f = 0 \end{aligned} \tag{8}$$

and that $\boldsymbol{x}_i^T$ is given by:

$$\boldsymbol{x}_i^T = [u_i^2, u_i v_i, v_i^2, u_i, v_i, 1] \tag{9}$$

the parameter vector of the ellipse that minimizes (7) based on N edge points for each of the objects of segmented is obtained.

In the minimization of (7), and in order to avoid the trivial solution $\boldsymbol{w} = 0$, a restriction should be imposed on $\boldsymbol{w}$. In this regard, of the multiple solutions possible, the one chosen is that in which the solution of (8) is an ellipse. This condition may be written as:

$$b^2 - 4ac = \boldsymbol{w}^T \cdot \boldsymbol{B} \cdot \boldsymbol{w} = -1 \tag{10}$$

with

$$\boldsymbol{B} = \begin{bmatrix} 0 & 0 & -2 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ -2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}. \tag{11}$$

Therefore, (7) may be written as:

$$d_w = \min_{\boldsymbol{w}} \|\boldsymbol{w}^T \cdot \boldsymbol{X}^T \cdot \boldsymbol{X} \cdot \boldsymbol{w}\| = \min_{\boldsymbol{w}} \|\boldsymbol{w}^T \cdot \boldsymbol{S} \cdot \boldsymbol{w}\| \tag{12}$$

$\boldsymbol{X}$ being the design matrix, which is given by:

$$\boldsymbol{X} = \begin{bmatrix} u_1^2 & u_1 v_1 & v_1^2 & u_1 & v_1 & 1 \\ u_2^2 & u_2 v_2 & v_2^2 & u_2 & v_2 & 1 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ u_N^2 & u_N v_N & v_N^2 & u_N & v_N & 1 \end{bmatrix} \tag{13}$$

and $\boldsymbol{S} = \boldsymbol{X}^T \cdot \boldsymbol{X}$ (scatter matrix). Via (12), the circumstance is reached in which the eigenvector associated with the only negative eigenvalue of $\boldsymbol{S}$ corresponds to the parameter vector associated with the ellipse that best matches N points $\boldsymbol{p}_i = [u_i, v_i]$, $i = 1, 2, \ldots, N$.

If, once this process has been finalized, there are r segmented objects within the window of interest, the algebraic distances are sorted in ascending order: $d_{w1} < d_{w2} < d_{w3} < d_{w4} < \cdots < d_{wr}$, and those distances whose d_{wi} is less than 0.5 pixels are selected as potential beacon elements. By identifying each of the objects segmented as m_{wi} and the selected group as $\boldsymbol{\Gamma}$, thus: $\boldsymbol{\Gamma} = \{m_{wi} : d_{wi} \leq 0.5\ pixels\}$.

In the next phase, those "estimated ellipses" whose eccentricity (E) exceeds a certain value and those whose relationship between the perimeter of the "estimated ellipse" and that of the edge points of the object used to obtain the corresponding ellipse (R_p) also exceed a certain value, are eliminated from the elements in group $\boldsymbol{\Gamma}$.

Eccentricity (E) of the ellipses is given by (14).

$$E = \sqrt{1 - \left(\frac{e_{jm}}{e_{jM}}\right)^2} \tag{14}$$

where e_{jm} and e_{jM} are respectively the minor and mayor axes of the ellipse in the image plane. The equation of the ellipse whose principal axes are equal to those of the image plane is given by (15):

$$\mu_1 \cdot u^2 + \mu_2 \cdot v^2 + [d \quad e] \cdot \boldsymbol{D} \cdot \begin{bmatrix} u \\ v \end{bmatrix} + f = 0 \tag{15}$$

where μ_1 and μ_2 are the eigenvalues of the matrix $\boldsymbol{M}$ given by (16) and $\boldsymbol{D} \in \Re^{2\times 2}$ is the eigenvector matrix corresponding to eigenvalues μ_1 and μ_2.

$$\boldsymbol{M} = \begin{bmatrix} a & b/2 \\ b/2 & c \end{bmatrix}. \tag{16}$$

From (14) and (15) the value of eccentricity is shown in the following expression:

$$E = \sqrt{1 - \left(\frac{\frac{1}{2}(\min(\frac{1}{\mu_1}, \frac{1}{\mu_2}))}{\frac{1}{2}(\max(\frac{1}{\mu_1}, \frac{1}{\mu_2}))}\right)}. \tag{17}$$

Relationship between perimeters, namely R_p, is defined as the quotient between the perimeters of the "estimated ellipse" identified by P_e and the contour of the object detected in the image, P_c, and which is used to estimate the ellipse:

$$R_P = \frac{Per.\text{“}estimated \quad ellipsen\text{”}}{Per.\text{“}boundary \quad points\text{”}} = \frac{P_e}{P_c} \tag{18}$$

where the "estimated ellipse" perimeter is approximated by:

$$P_e \approx \pi(\mu_a + \mu_b)\left(1 + \frac{3(\frac{\mu_a - \mu_b}{\mu_a + \mu_b})^2}{10 + \sqrt{4 - 3(\frac{\mu_a - \mu_b}{\mu_a + \mu_b})^2}}\right) \tag{19}$$

where μ_a and μ_b are respectively the lengths of the minor and major axes of the ellipse.

Thus, after this process, and by sorting the estimated ellipses in order of ascending eccentricity: $E_1 < E_2 < E_3 < E_4 < \cdots$, the potential elements of the beacon, OE_i, will be the group $\boldsymbol{\Pi} = \{OE_1, OE_2, OE_3, OE_4\}$ associated with the four with the lowest eccentricity. Likewise, by sorting the values of the relationship between perimeters in descending order: $R_{p1} > R_{p2} > R_{p3} > R_{p4}$, the potential beacons, OP_i, will be the group, $\boldsymbol{\eta} = \{OP_1, OP_2, OP_3$ and $OP_4\}$ associated with the four highest relationships between perimeters.

As a consequence, the potential beacons will be those objects in which OE_1, OE_2, OE_3, and OE_4 coincide with OP_1, OP_2, OP_3, and OP_4. If coincidence exists between the two groups, the next step is to verify that the four objects fulfill the geometric relationships for the relative physical location between the four elements that make up the beacon. In the case of either there being no coincidence between the groups of four objects $\{OE_1, OE_2, OE_3$ and $OE_4\} \neq \{OP_1, OP_2, OP_3$ and $OP_4\}$, or in the case of the geometric position relationships between them not being fulfilled, the window of interest will be enlarged and the process will be repeated. If the beacon is still not detected even when the entire image is analyzed, a new image will be captured and analyzed. Figure 7 shows the values of the parameters d_w, E and R_p for each of the "estimated ellipses" in the image in Fig. 6.

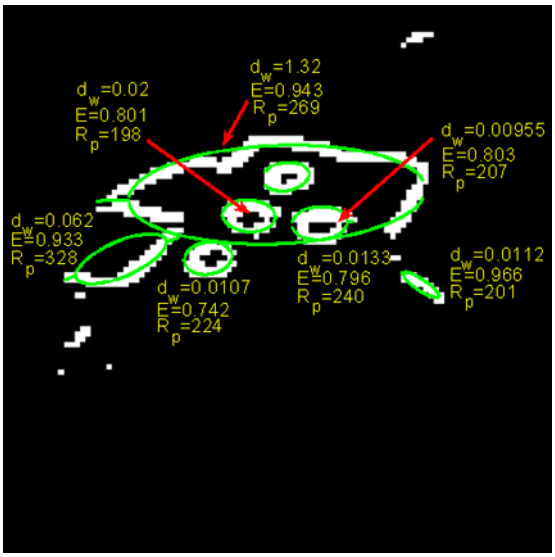


Fig. 7 Beacon detection example. Eight objects are shown and the parameters d_w, E and R_p are indicated for each of the "estimated ellipses" associated with each of them. As may be observed, the elements belonging to the beacon are easily identifiable

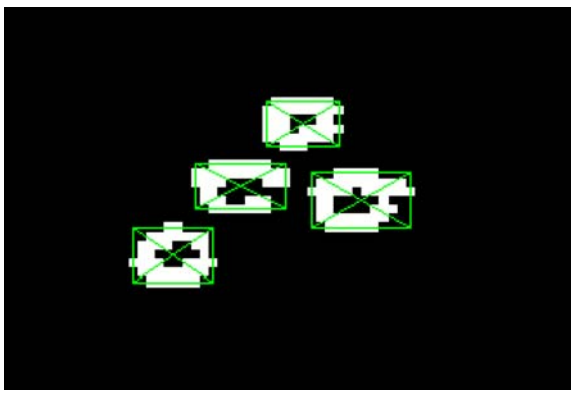


Fig. 8 Example of obtaining the coordinates of the centroid of each of the elements that make up the beacon

3.4 Obtaining the centroid of the beacon elements

After segmenting and identifying the beacon, the next step consists of obtaining the pixel coordinates of each of its centroids (u_c, v_c). In our case, these coordinates are obtained at the point of intersection of the main diagonal lines that link the vertices of the minimum-sized rectangle that contains each of the beacon's elements (Fig. 8).

4 Positioning Algorithm

Once the centroids of the beacons have been obtained in the images captured by each of the cameras that make up the vision system, the next step is to obtain the robot's 3D position. Although the robot may be captured by either one or several cameras, in this paper reference will fundamentally be made to the most frequently encountered situation, in which the robot is captured by two cameras. In the case that the robot is within the field of view of more than two cameras, the process followed is similar to that applied when the robot is captured by two. Nevertheless, the authors of this paper have been able to verify in various experiments that using more than two cameras to detect the robot's position does not represent a notable improvement on the results achieved with just two cameras. On the other hand, the case of a single camera detecting the robot is only valid when the robot moves on one plane (known Z coordinates). As a result, this paper will only make a brief reference to the results obtained using a single camera.

4.1 Obtaining the robot's position using two cameras

Firstly, it should be indicated that the position of the centroid of element C of the beacon will be taken as the robot's 3D position. The coordinates of the centroids (the point that we may identify as $\boldsymbol{P}$ in the 3D environment) of any of the elements (A, B, C, D) that make up the beacon are obtained from the projection lines that extend from the optic centers (Ψ_i, Ψ_j) of each camera to said point (centroid of each element), and include their projection on the image planes $(\boldsymbol{p}_i, \boldsymbol{p}_j)$. As a consequence, the 3D coordinates of the centroid of an element of the beacon will be the point of intersection of these projection lines. Unfortunately, the different sources of error that exist throughout the process (detection process, calibration, etc.) mean that the projection lines are not truncated and cross over. Therefore, it is necessary to propose a solution that minimizes the error in the estimation of point $\boldsymbol{P}$ (centroid of one of the elements of the beacon) generated from the two intersecting projection lines.

In order to resolve this problem, the authors of this paper propose as point $\boldsymbol{P}$ that point that minimizes the sum of the Mahalanobis distances between said point $\boldsymbol{P}$ and the projection lines. As a result, obtaining point $\boldsymbol{P}$ will be a function of the errors produced in obtaining the pixel coordinates from the cameras, giving greater protagonism to the camera that produces least error.

By using a "pin-hole" perspective projection model given by (4), after identifying the 3D coordinates of point $\boldsymbol{P}$, referred to the global reference system, as $[X, Y, Z]^T$, the relationship between these coordinates and their corresponding pixel coordinates in the cameras $i(u_i, v_i)$ and $j(u_j, v_j)$, on the image plane of each camera, is given by (20) and (21).

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \lambda_i \cdot R_i' \cdot \begin{bmatrix} \frac{u_i - u_{oi}}{f_{xi}} \\ \frac{v_i - v_{oi}}{f_{yi}} \\ 1 \end{bmatrix} + T_i', \tag{20}$$

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \lambda_j \cdot R'_j \cdot \begin{bmatrix} \frac{u_j - u_{oj}}{f_{xj}} \\ \frac{v_j - v_{oj}}{f_{yj}} \\ 1 \end{bmatrix} + T'_j \tag{21}$$

where $R'_k = R_k^{-1}$ and $T'_k = -R_k^{-1} \cdot T_k$, $\boldsymbol{R}_k$ and $\boldsymbol{T}_k$ being the rotation and translation matrices of camera $k(k = i, j)$, referred to the global reference system, u_k, v_k being the pixel coordinates of the centroid of the projection of the element of the beacon in the image and λ_i and λ_j being scale factors.

Based on (20) and (21), it is possible to write:

$$\begin{aligned} \boldsymbol{P}_i &= \lambda_i \cdot \boldsymbol{R}_i^{-1} \cdot \begin{bmatrix} \frac{u_i - u_{oi}}{f_{xi}} \\ \frac{v_i - v_{oi}}{f_{yi}} \\ 1 \end{bmatrix} - \boldsymbol{R}_i^{-1} \cdot \boldsymbol{T}_i, \\ \boldsymbol{P}_j &= \lambda_j \cdot \boldsymbol{R}_j^{-1} \cdot \begin{bmatrix} \frac{u_j - u_{oj}}{f_{xj}} \\ \frac{v_j - v_{oj}}{f_{yj}} \\ 1 \end{bmatrix} - \boldsymbol{R}_j^{-1} \cdot \boldsymbol{T}_j. \end{aligned} \tag{22}$$

In the presence of error free values in the pixel coordinates (u_i, v_i and u_j, v_j), points $\boldsymbol{P}_i$ and $\boldsymbol{P_j}$ given by (22) must be equal ($\boldsymbol{P}_i = \boldsymbol{P}_j$), and therefore correspond to the point of intersection of the projection lines. As a result:

$$\begin{aligned} &\lambda_i \cdot \boldsymbol{R}_i^{-1} \cdot \begin{bmatrix} \frac{u_i - u_{oi}}{f_{xi}} \\ \frac{v_i - v_{oi}}{f_{yi}} \\ 1 \end{bmatrix} - \boldsymbol{R}_i^{-1} \cdot \boldsymbol{T}_i \\ &\quad = \lambda_j \cdot \boldsymbol{R}_j^{-1} \cdot \begin{bmatrix} \frac{u_j - u_{oj}}{f_{xj}} \\ \frac{v_j - v_{oj}}{f_{yj}} \\ 1 \end{bmatrix} - \boldsymbol{R}_j^{-1} \cdot \boldsymbol{T}_j. \end{aligned} \tag{23}$$

Expression (23) can be rewritten by:

$$\boldsymbol{A} \cdot \boldsymbol{B} - \boldsymbol{T} = 0 \tag{24}$$

where $\boldsymbol{A}$ is a 3×2 matrix, $\boldsymbol{B}$ is a 2×1 matrix, and $\boldsymbol{T}$ is a 3×1 matrix, which are given by:

$$\begin{aligned} &\boldsymbol{A} = \begin{bmatrix} -\boldsymbol{R} \cdot \begin{bmatrix} \frac{u_j - u_{oj}}{f_{xj}} \\ \frac{v_j - v_{oj}}{f_{yj}} \\ 1 \end{bmatrix}, \begin{bmatrix} \frac{u_i - u_{oi}}{f_{xi}} \\ \frac{v_i - v_{oi}}{f_{yi}} \\ 1 \end{bmatrix} \end{bmatrix}; \qquad \boldsymbol{B} = \begin{bmatrix} \lambda_j \\ \lambda_i \end{bmatrix}; \\ &\boldsymbol{T} = \boldsymbol{T}_i - \boldsymbol{R} \cdot \boldsymbol{T}_j \end{aligned} \tag{25}$$

such that $\boldsymbol{R} = \boldsymbol{R}^i \cdot \boldsymbol{R}_j^{-1}$.

By using a least squares approach to solve (24) the following close expression is obtained:

$$\begin{bmatrix} \lambda_j \\ \lambda_i \end{bmatrix} = (A^T \cdot A)^{-1} \cdot A^T \cdot T. \tag{26}$$

If pixel coordinate values are contaminated by noise, then the point coordinates $\boldsymbol{P}_i$ and $\boldsymbol{P_j}$ given in (22) are not coincident ($\boldsymbol{P}_i \neq \boldsymbol{P}_j$). In that case a possible solution is to assign point $\boldsymbol{P}$ the coordinates of the midpoint of the segment that joins $\boldsymbol{P}_i$ and $\boldsymbol{P}_j$ (Fernández et al. 2003a). Furthermore, this is valid if the condition is met that the errors produced by both cameras in obtaining the centroids of the elements of the beacon are similar. However, in practice this does not occur due to various circumstances, one of the most noteworthy of which is the difference in distance between the beacon and the two cameras. Bearing these considerations in mind, the authors of this paper propose to assign $\boldsymbol{P}$ the value that results from minimizing the cost function, χ, defined in (27).

$$\chi = \sum_{k=i,j} (\boldsymbol{P} - \boldsymbol{P}_k)^T \cdot \boldsymbol{C}_k^{-1} \cdot (\boldsymbol{P} - \boldsymbol{P}_k) \tag{27}$$

where $\boldsymbol{C_k}$ $(k = j, i)$ represent the covariance matrices associated with the coordinates of $\boldsymbol{P_k}$ $(k = j, i)$, and which is given by:

$$\boldsymbol{C}_k = \begin{bmatrix} \sigma^2_{X(k)} & \rho_{XY(k)}\sigma_{X(k)}\sigma_{Y(k)} & \rho_{XZ(k)}\sigma_{X(k)}\sigma_{Z(k)} \\ \rho_{XY(k)}\sigma_{Y(k)}\sigma_{X(k)} & \sigma^2_{Y(k)} & \rho_{YZ(k)}\sigma_{Y(k)}\sigma_{Z(k)} \\ \rho_{XZ(k)}\sigma_{Z(k)}\sigma_{X(k)} & \rho_{YZ(k)}\sigma_{Y(k)}\sigma_{Z(k)} & \sigma^2_{Z(k)} \end{bmatrix} \tag{28}$$

where $\sigma^2_{X(k)}$, $\sigma^2_{Y(k)}$ and $\sigma^2_{Z(k)}$ represent the variances of the coordinates X_k, Y_k, Z_k, of points $\boldsymbol{P}_k$ $(k = j, i)$, and $\rho_{XY(k)}$, $\rho_{XZ(k)}$ and $\rho_{YZ(k)}$ represent the correlation coefficient between coordinates.

Obtaining the minimum value of (27), and assuming that the measurements of X_k, Y_k, Z_k $(k = j, i)$ are uncorrelated, the coordinates of point $\boldsymbol{P}$ are given by:

$$\begin{aligned} X &= \frac{\sigma^2_{Xi} X_i + \sigma^2_{Xj} X_j}{\sigma^2_{Xi} + \sigma^2_{Xj}}, \\ Y &= \frac{\sigma^2_{Yi} Y_i + \sigma^2_{Yj} Y_j}{\sigma^2_{Yi} + \sigma^2_{Yj}}, \\ Z &= \frac{\sigma^2_{Zi} Z_i + \sigma^2_{Zj} Z_j}{\sigma^2_{Zi} + \sigma^2_{Zj}}. \end{aligned} \tag{29}$$

As may be verified, if $\sigma^2_{Xi} = \sigma^2_{Xj}$, $\sigma^2_{Yi} = \sigma^2_{Yj}$, $\sigma^2_{Zi} = \sigma^2_{Zj}$, then the coordinates of point $\boldsymbol{P}$ correspond to the midpoint of the segment that joins $\boldsymbol{P}_i$ and $\boldsymbol{P}_j$.

Once the values of the coordinates of point $\boldsymbol{P}$ have been obtained, depending on the variances of the 3D coordinates, it only remains to express these according to the variances of the pixel coordinates. In short, it is a case of expressing variances $\sigma^2_{X(k)}$, $\sigma^2_{Y(k)}$ and $\sigma^2_{Z(k)}$ according to the variances of coordinates (u_i, v_i) and (u_j, v_j), σ^2_{ui}, σ^2_{vi}, σ^2_{uj} and σ^2_{vj}.

For this purpose, expressing (22) as is indicated in (29), r_{ij} being the coefficients of matrix $\boldsymbol{R}$:

$$\begin{aligned}\boldsymbol{P}_k &= \begin{bmatrix} X_k \\ Y_k \\ Z_k \end{bmatrix} \\ &= \lambda_{(k)} \begin{bmatrix} r_{11(k)} & r_{21(k)} & r_{31(k)} \\ r_{12(k)} & r_{22(k)} & r_{32(k)} \\ r_{13(k)} & r_{23(k)} & r_{33(k)} \end{bmatrix} \cdot \begin{bmatrix} \frac{u_k - u_{0(k)}}{f_{x(k)}} \\ \frac{v_k - v_{0(k)}}{f_{y(k)}} \\ \\ 1 \end{bmatrix} \\ &\quad + \begin{bmatrix} T'_{X(k)} \\ T'_{Y(k)} \\ T'_{Z(k)} \end{bmatrix} \end{aligned} \tag{30}$$

and deriving (30) as regards $\lambda_{(k)}$, u_k and v_k, it is possible to obtain:

$$\begin{aligned}\sigma^2_{X(k)} &= \boldsymbol{J}_{X(k)} \cdot \begin{bmatrix} \sigma^2_{\lambda(k)} & 0 & 0 \\ 0 & \sigma^2_{u(k)} & 0 \\ 0 & 0 & \sigma^2_{v(k)} \end{bmatrix} \cdot \boldsymbol{J}^T_{X(k)}, \\ \sigma^2_{Y(k)} &= \boldsymbol{J}_{Y(k)} \cdot \begin{bmatrix} \sigma^2_{\lambda(k)} & 0 & 0 \\ 0 & \sigma^2_{u(k)} & 0 \\ 0 & 0 & \sigma^2_{v(k)} \end{bmatrix} \cdot \boldsymbol{J}^T_{Y(k)}, \\ \sigma^2_{Z(k)} &= \boldsymbol{J}_{Z(k)} \cdot \begin{bmatrix} \sigma^2_{\lambda(k)} & 0 & 0 \\ 0 & \sigma^2_{u(k)} & 0 \\ 0 & 0 & \sigma^2_{v(k)} \end{bmatrix} \cdot \boldsymbol{J}^T_{Z(k)} \end{aligned} \tag{31}$$

where $\boldsymbol{J}_{X(k)}$, $\boldsymbol{J}_{Y(k)}$, $\boldsymbol{J}_{Z(k)}$ represent the Jacobians of coordinates X_k, Y_k and Z_k, respectively.

Finally, in order to obtain the values of $\sigma^2_{\lambda i}$ and $\sigma^2_{\lambda j}$, (26) is derived with respect to u_i, v_i and u_j, v_j, λ_i, and λ_j, obtaining:

$$\begin{aligned}&\begin{bmatrix} \sigma^2_{\lambda i} & \rho_{\lambda i \lambda j} \cdot \sigma_{\lambda i} \cdot \sigma_{\lambda j} \\ \rho_{\lambda i \lambda j} \cdot \sigma_{\lambda i} \cdot \sigma_{\lambda j} & \sigma^2_{\lambda j} \end{bmatrix} \\ &= \boldsymbol{J}_1^+ \cdot \boldsymbol{J}_2 \cdot \begin{bmatrix} \sigma^2_{ui} & 0 & 0 & 0 \\ 0 & \sigma^2_{uj} & 0 & 0 \\ 0 & 0 & \sigma^2_{vi} & 0 \\ 0 & 0 & 0 & \sigma^2_{vj} \end{bmatrix} \cdot \boldsymbol{J}_1^T \cdot \boldsymbol{J}_1^+ \end{aligned} \tag{32}$$

where $\boldsymbol{J}_1^+$ represents the pseudoinverse of $\boldsymbol{J}_1$, the value of which is given by (33):

$$\boldsymbol{J}_1^+ = \boldsymbol{J}_1^T \cdot (\boldsymbol{J}_1 \cdot \boldsymbol{J}_1^T)^{-1}. \tag{33}$$

As $\boldsymbol{J}_1 \cdot \boldsymbol{J}_1^T$ is singular, it is possible to obtain (32) using singular value decomposition (SVD).

In order to verify the effectiveness of the proposed method, every test was carried out in a rectangular 6 × 4 m. environment in which four SONY XC-ST70CE equipped by a CCD of 2/3″ and a 8.5 mm optic were located. Image resolution is around 576 × 768 pixels with 8-bit greyscale information. Camera synchronization is carried out by a trigger signal generated by one of the cameras, configured as a master. Tests include natural and artificial illumination conditions, from low luminosity to high bright scenes. The acquisition system is composed by a Matrox Multichaner, and its processing software under Windows O.S. The robot was placed in different positions within the environment and in each of these a total of 10^3 measurements of its position were taken. The mean positioning error is given by

$$e_p = \frac{1}{10^3} \sum_{k=1}^{10^3} |\boldsymbol{P}_{k(\text{actual})} - P_{k(\text{measurement})}|. \tag{34}$$

Figure 9(a) shows this error (e_p) for the robot's positions drawn in Fig. 5(a), while Fig. 9(b) shows the standard deviation (σ_{ep}) for each position. As may be observed, the results are highly satisfactory as the position error does not exceed 40 mm and the standard deviation does not exceed 9 mm in any of the cases.

4.2 Obtaining the robot's position using a single camera

Assuming that the robot moves along a known and constant Z coordinate plane (Z_0), it is possible to obtain the 3D coordinates using a single camera. These may be obtained by using (30) and substituting Z_k for Z_0.

5 Guidance System

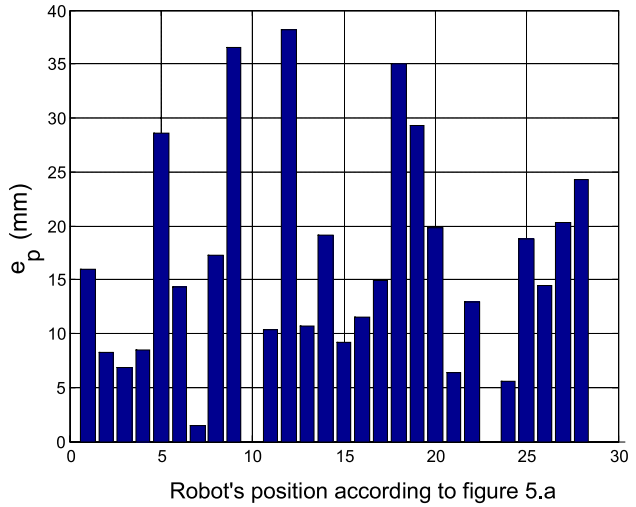
Figure 10 shows the general structure of the control system used, indicating the most important modules responsible for carrying out the various tasks associated with guidance of the robot.

As may be observed in Fig. 10, two main blocks have been identified—the guidance module and the local positioning system (LPS). The first of these is responsible for generating the velocity set-points to be sent to the robot, whilst the second is responsible for obtaining the robot's position from the sensor system.

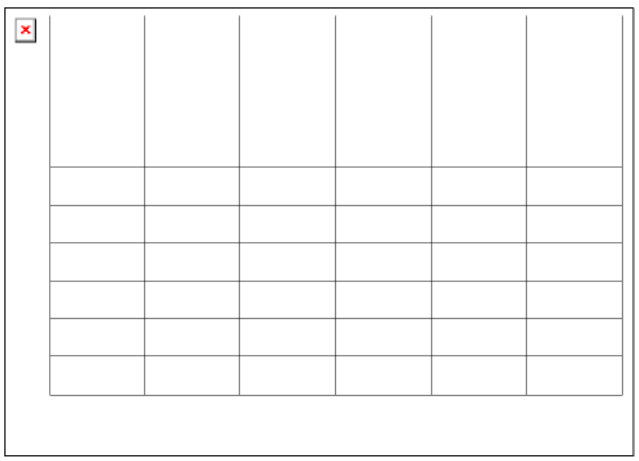
It is also worth pointing out that in the system implemented two types of sensors have been used to create the guidance system—cameras located within the work environment that provide the robot's absolute position, and encoders (odometry).

Although the odometry system should be used only in those trajectory paths where the robot is not observed by any camera, a Kalman Filter is used at each frame to fuse information between the odometry system and the localization given by the cameras. Using the extra information from

odometry, it can relax the conditions imposed to the vision system. The Kalman Filter provides a way to obtain an optimal estimation of a robot's pose by taking into account the noise and uncertainty present in the visual system and the odometry readings. Evidently, the use of odometry may be avoided by incorporating a sufficient number of cameras in the environment (ensuring that the robot is always within the field of view of at least one camera, in the case of flat surfaces, or at least two in other cases), and reducing the processing time (i.e. by using dedicated hardware). The tests carried out, because of the reduced number of cameras and the dimensions of the room (6 meters width and 4 meters large) and a hall of (14 meters width and 1.5 meters large), had areas in which only two cameras were observing the robot, others observed by only one camera and also ones without observation of the robot. The later situation is solved by trusting in the odometry system.



(a) Mean value of the position error



(b) Standard deviation of the position error

Fig. 9 Real robot position errors using two cameras to obtain the robot's position

Implementation of the guidance model is based on a map of the environment, navigation task specifications (origin and destination points) and the robot's current position. Using this data, the trajectory or path that the robot should follow is calculated. The set-point generator is responsible for generating the velocities of the robot's drive wheels so that the robot follows the desired trajectory.

Cubic spline curves (Fernández et al. 2003b) have been used in the trajectory generation process. Given that this type of curve cannot present two different ordinates for the same abscissa (due to its condition as a real mathematical function), this represents a problem that may be resolved using various solutions (Lázaro 1998). In our case, we have opted to make use of parameterized curves because of their simplicity and ease of calculation. In the case of flat surfaces, the parameterized curves represent a function $\boldsymbol{Q}(\Lambda)$ on a bidimensional plane *(X, Y)* in accordance with an advance parameter denominated "Λ":

$$\boldsymbol{Q}(\Lambda) = (f_X(\Lambda), f_Y(\Lambda)) \tag{35}$$

where $f_X(\Lambda)$ and $f_Y(\Lambda)$ represent the variation of the abscissas and ordinates, respectively, in accordance with

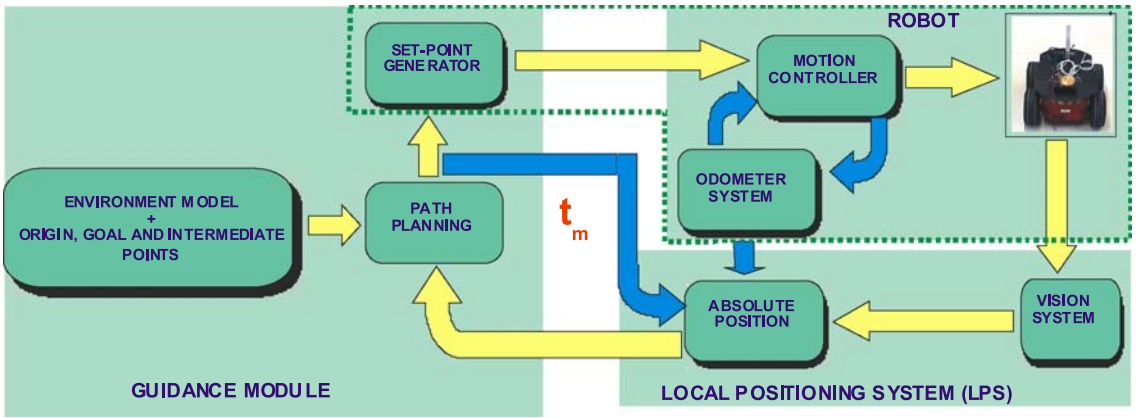


Fig. 10 Block diagram of the control system used

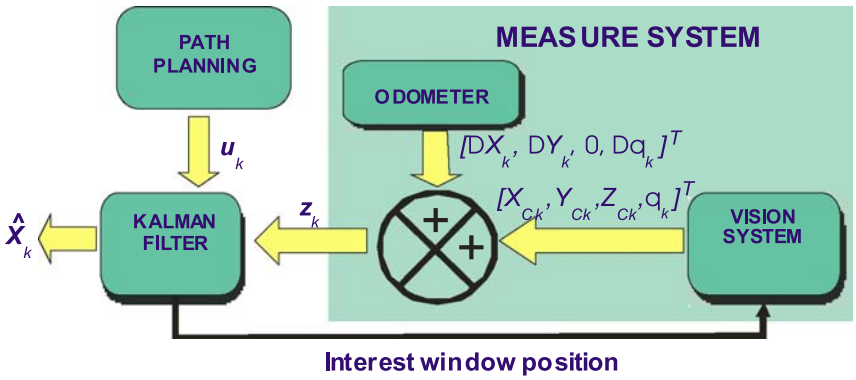


Fig. 11 Structure of the positioning system

advance parameter "Λ". In order to implement functions $f_X(\Lambda)$ and $f_Y(\Lambda)$, third degree polynomials are used. Taking into account that it is possible to define as many intermediate points in the path as necessary, the curve is divided into sections (one for every two consecutive points) and each of these is defined by a polynomial. As a consequence, if there are n number of $\boldsymbol{P_i}$ travel points, with coordinates $[X_i, Y_i]^T (i = 0, 1, \ldots, n-1)$, the spline will have a total of $n-1$ sections ($\boldsymbol{Q}_0, \boldsymbol{Q}_1, \ldots, \boldsymbol{Q}_{n-2}$). Thus, section i will be defined by two polynomials of the type:

$$\begin{aligned} f_{Xi}(\Lambda) &= a_{iX} + b_{iX}\Lambda + c_{iX}\Lambda^2 + d_{iX}\Lambda^3, \\ f_{Yi}(\Lambda) &= a_{iY} + b_{iY}\Lambda + c_{iY}\Lambda^2 + d_{iY}\Lambda^3 \end{aligned} \tag{36}$$

with $0 \leq \Lambda \leq 1$ and $(a_{iX}, b_{iX}, c_{iX}, d_{iX})$ and $(a_{iY}, b_{iY}, c_{iY}, d_{iY})$ being the coefficients of the abscissas' and ordinates' curve, respectively, for each section of the trajectory. In order to calculate the coefficients of (36) in each section, the conditions of continuity in the first and second derivative in all of the intermediate points of the spline are imposed, ensuring moreover an orientation (first derivative) of the curve for the end points of the same (origin and destination).

Taking into account that the measurements provided by the vision system and by odometry are contaminated by noise, a Kalman Filter (Welch and Bishop 2001) (Fig. 11) has been used to minimize this effect. Moreover, at instant k this allows the robot's position at $k+1$ to be estimated and, as a result, also allows the position of the window of interest, to which the vision algorithms are applied, to be estimated.

The robot platform used in the tests include a built-in low level control loop whose inputs are references in the linear speed achieved on each robot wheel (V_L, V_R). In this case the required reference of speed, namely $\upsilon_k = [V_{Lk}, V_{Rk}]$, is obtained from the present robot's pose given by the Kalman estimation and the next pose to reach, which is given by the path to follow (37).

If robot path is in a plane, the pose of the robot can be completely described by three components $[X_{k-1}, Y_{k-1}, \theta_{k-1}]$, where X_k and Y_k represent the coordinates of the center of rotation of the robot (located in the mid point of the axis between both wheels in a differential robot) and θ_k is the orientation of the robot in the plane (Fig. 12), the linear speed in the plane $\dot{X}_K, \dot{Y}_K$ and the angular sped $\dot{\theta}_K$ from the center of rotation of the robot in the XY plane are given by:

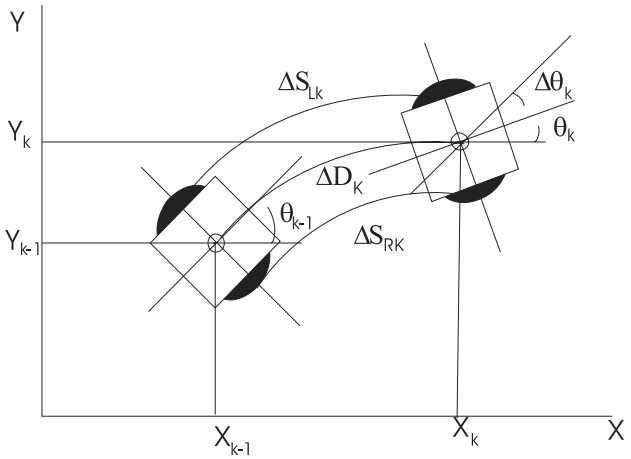


Fig. 12 Robot motion representation between two consecutives time samples

$$\begin{aligned} \dot{X}_k &= \frac{V_{Rk} + V_{Lk}}{2} \cdot \cos(\theta_k) = V_k \cdot \cos(\theta_k), \\ \dot{Y}_k &= \frac{V_{Rk} - V_{Lk}}{2} \cdot \sin(\theta_k) = V_k \cdot \sin(\theta_k), \\ \dot{\theta}_k &= \frac{V_{Rk} - V_{Lk}}{B} \end{aligned} \tag{37}$$

where B *is the distance between the wheels.*

By expressing as ΔS_R and ΔS_L the linear increments in the right and left wheels, increments in position (ΔD_k) and orientation $(\Delta\theta_k)$ are obtained for the interval t_m (interval between the generation of two consecutive speed references):

$$\begin{aligned} \Delta D_k &= \frac{\Delta S_{Rk} + \Delta S_{Lk}}{2} = t_m \frac{V_{Rk} + V_{Lk}}{2}, \\ \Delta\theta_k &= \frac{\Delta S_{Rk} - \Delta S_{Lk}}{B} = t_m \frac{V_{Rk} - V_{Lk}}{B}. \end{aligned} \tag{38}$$

By integration of (38) the position of the robot can be approximated in k from the position at $k-1$:

$$X_k = X_{k-1} + \Delta D_k \cos\left(\frac{\theta_k + \theta_{k-1}}{2}\right),$$
$$Y_k = Y_{k-1} + \Delta D_k \sin\left(\frac{\theta_k + \theta_{k-1}}{2}\right), \quad (39)$$
$$\theta_k = \theta_{k-1} + \Delta\theta_k.$$

From (39) the speed reference vector υ_k that is dispatched to robot control is obtained

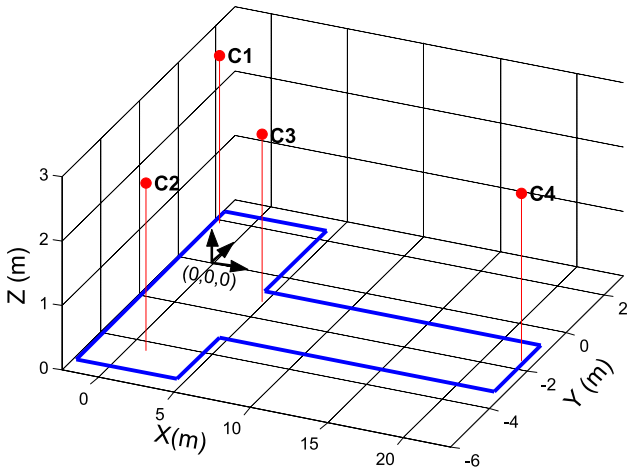
$$\upsilon_k = \frac{1}{t_m}\begin{pmatrix} \frac{X_k - X_{k-1}}{\cos(\frac{\theta_k+\theta_{k-1}}{2})} - \frac{B\cdot(\theta_k-\theta_{k-1})}{2} \\ \frac{X_k - X_{k-1}}{\cos(\frac{\theta_k+\theta_{k-1}}{2})} + \frac{B\cdot(\theta_k-\theta_{k-1})}{2} \end{pmatrix}. \quad (40)$$

6 Results

Practical implementation of this system employed four monochrome Sony XC-ST70CE cameras with an 8.5 mm optic and a Pentium II 400 MHz computer with Matrox Meteor2-MC frame grabber cards. Under these conditions, the time taken to capture and process the images by the four cameras was 0.6 s. Time between two consecutive speed references sent to the robot is closely the same than the acquisition and processing time. The remaining tasks involved in the control process can be approximated to be instantaneous. The robot's mean travel speed was 0.3 m/s.

In order to verify the effectiveness of the proposed system, numerous practical tests have been carried out within the building of the Polytechnic School of the University of Alcala. The environment used has consisted of a 4×6 m room and a 14×1.5 m corridor (Fig. 13), in which the cameras (C1, C2, C3 and C4) were located at coordinates (-1.8 m, 1.8 m, 2.6 m), (1.1 m, -4.4 m, 2.6 m), (4.8 m, -1.3 m, 2.5 m) and (22.1 m, -1.6 m, 2.5 m), respectively, the coordinate origin being that indicated in Fig. 13. Figure 13(b) shows the location of camera C4 used in the practical tests. It must be pointed out that, despite all tests have been done in planar environments, the solution proposed can also be used in environments where the robot moves in 3D, as those equipped with different heights and access ramps for wheelchairs. For future work, these kinds of experiments are planned to be used.

Test 1: In this test, a straight trajectory was followed. The origin coordinates were (P_s) $X_s = 0$, $Y_s = -0.4$ m and the destination coordinates were $X_e = 3.2$ m $Y_e = -0.4$ m. Figure 14 shows the trajectory followed by the robot. The coordinates of the destination reached were $X = 3.1936$ m $Y = -0.3985$ m. As may be observed, the greatest position error is produced at the start of the trajectory and this is due to the error produced when manually positioning the robot.



(a) Work environment and camera distribution used in the practical tests

(b) Location of camera C4

Fig. 13 Environment used in the tests

Test 2: In this test, a quadrilateral trajectory consisting of four straight segments was selected, on which the robot is within the field of view of two cameras at all times. Figure 15 shows the preset points through which the robot should pass and the real trajectory followed by the same. Table 1 shows the coordinates of the vertices (X_v, Y_v) through which the robot should pass and the real coordinates (X_a, Y_a) of the points through which the robot actually passed.

Test 3: In this test, the robot was guided through the environment consisting of the room and the corridor (Fig. 13), using four cameras (C1, C2, C3 and C4). In this case, the trajectory that the robot should follow contains some sections that are within two cameras' field of view, some of that are within a single camera's field of view, and some sections are not within any camera's field of view. In this latter case, odometry is employed to perform guidance of the robot. Figure 16(a) shows the trajectory that the robot should follow and the trajectory that it actually followed. The figure shows four trajectory points through which the robot should pass

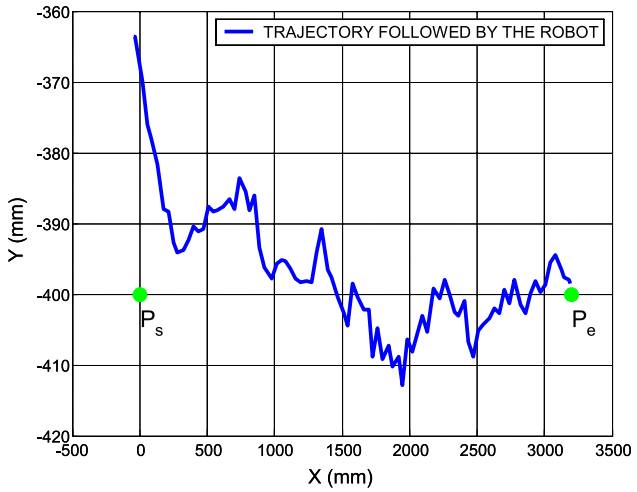


Fig. 14 Trajectory followed by the robot in the first practical test

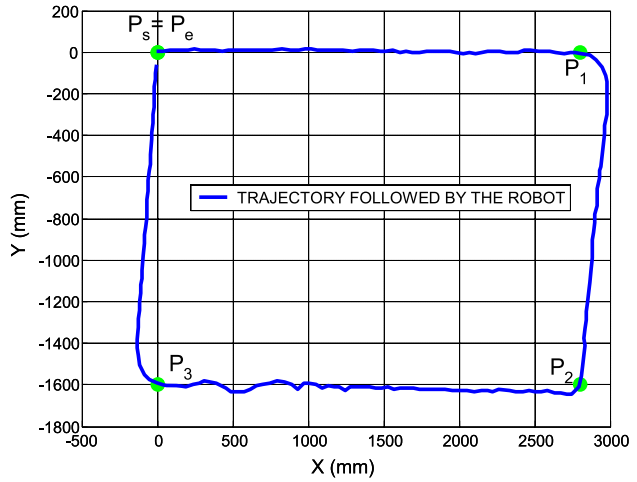
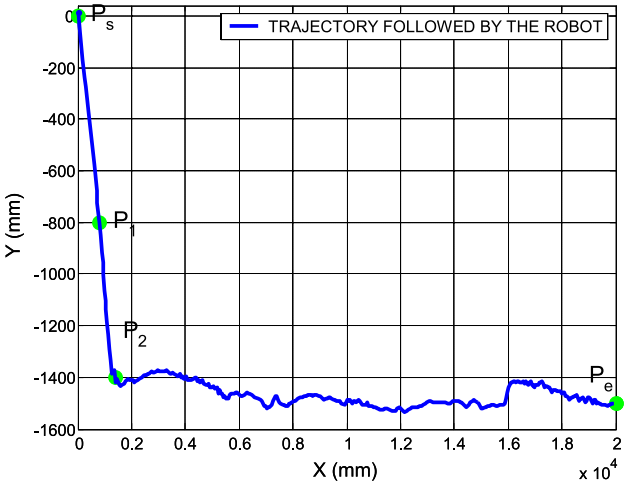


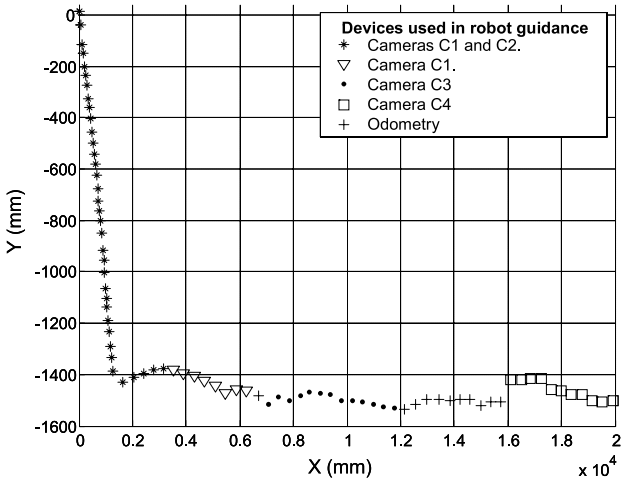
Fig. 15 Trajectory followed by the robot in the second practical test

Table 1 Coordinates, in meters, of the trajectory points that the robot should pass through in test 2

	Coordinates of the intended trajectory points (m)		Coordinates of the actual trajectory points (m)	
	X_v	Y_v	X_a	Y_a
P_S	0	0	0.04	0.005
P_1	2.8	0	2.7916	0.0008
P_2	2.8	−1.6	2.7971	−1.5728
P_3	0	−1.6	0.0003	−1.5978
P_e	0	0	−0.006	−0.051



(a) Trajectory followed



(b) Devices (cameras and odometry) used to obtain the robot's position on different sections

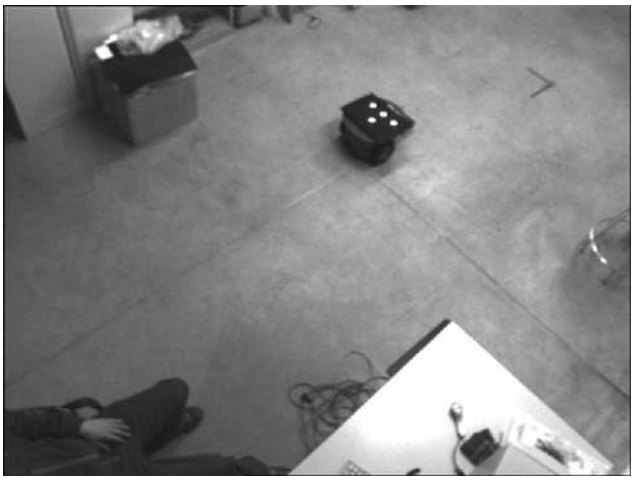
Fig. 16 Example of a practical experiment carried out in a large-size environment using four cameras

Table 2 Coordinates, in meters, of the trajectory points that the robot should pass through in test 3

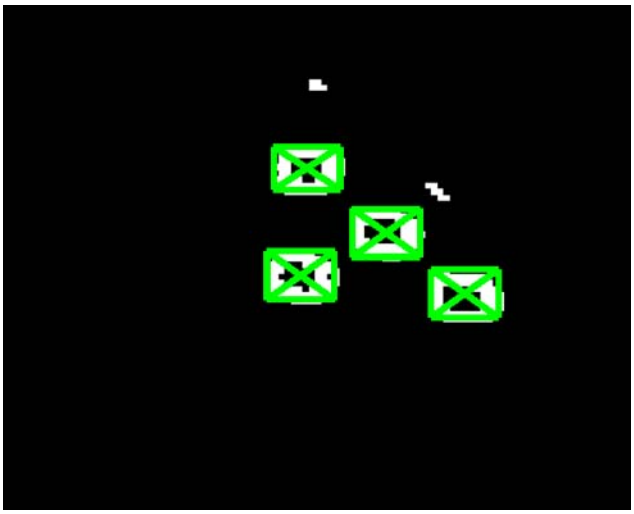
	Coordinates of the intended trajectory points (m)		Coordinates of the actual trajectory points (m)	
	X_v	Y_v	X_a	Y_a
P_S	0	0	0.01	0.02
P_1	0.8	−0.8	0.801	−0.803
P_2	1.4	−1.4	1.408	−1.222
P_e	20	−1.5	19.91	−1.502

(P_s, P_1, P_2, P_e), the coordinates for which are shown in Table 2. Figure 16(b) shows the actual trajectory followed by the robot and also indicates the number of cameras that capture the robot in each section.

Finally, in order to show the level robustness of the tracking system of the beacon against the presence of different lighting conditions and light spots, Fig. 17 shows the result with following lighting conditions: direct natural lighting,

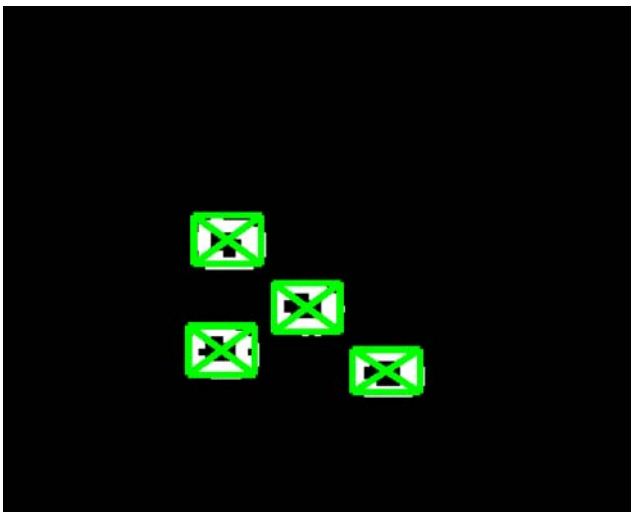


(a) Artificial lighting



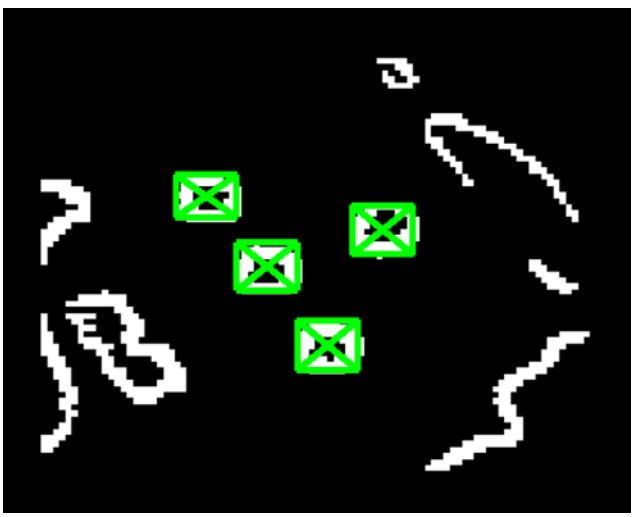
(b) Beacon detection in (a)

(c) Low ambient illumination



(d) Beacon detection in (c)

(e) Natural lighting and brightness



(f) Beacon detection in (e)

Fig. 17 Pattern detection under different lighting conditions

artificial lighting, and presence of spots originated by natural lighting.

7 Conclusions

This paper has presented an alternative guidance system for mobile robots using cameras located within the movement environment. In the proposed solution, infrared beacons mounted on board the robots have been used in order to facilitate both detection of the robots and accurate capture of the robots' 3D coordinates. In this paper, the importance of taking into account the various types of error that are generated by this type of proposed mobile robot guidance system has been made clear. Among these, those arising from the type of camera and optics used, as well as from the image capture and processing hardware, stands out particularly. Another significant aspect is obtaining the robot's 3D coordinates and orientation from the coordinates on the image plane of the centroids of the four elements that make up the IR beacon. The proposal made in this paper has demonstrated the feasibility of carrying out guidance based on the information provided by cameras located within the robots' movement environment and, moreover, it has produced position errors in the order of millimeters. Furthermore, the proposed solution offers a significant advantage of minimizing the electronic and sensor systems which are mounted on board the robots and also offers the possibility of using the same infrastructure to carry out guidance of several robots. The various practical tests carried out in real environments have been highly satisfactory, in spite of the limitations of the image processing and control hardware used (as well as the limited number of cameras). They have made clear that the proposal is feasible without the need to use a high number of cameras. It has been verified that the most advisable practical solution (taking into account the cost of the infrastructure) is for the trajectory that the robot should follow to be captured by two cameras at all times. It is also worth pointing out that in environments with flat surfaces it is possible to perform guidance using a single camera at all times and, in those sections of the trajectory where it is difficult for the robot to be captured by the cameras, it is possible to make use of odometry (provided that said sections are of a limited size).

The authors of this paper are currently working on obtaining the robot's position using natural landmarks on the same (Pizarro et al. 2005) (without the need to use artificial beacons on board the robots).

Acknowledgements This work was supported by the Ministry of Science and Technology under RESELAI and SILPAR II projects (references TIN2006-14896-C02-01 and DPI2006-05835).

References

AICON 3D. (Online). Available: http://www.aicon.de/.

Borestein, J., Everet, H. R., & Feng, L. (1996). *Where am I? Sensor and methods for mobile robot positioning*. The University of Michigan. Available: http://www-personal.umich.edu/~johannb/shared/pos96rep.pdf.

Chung, J., Kim, N., Kim, J., & Park, C. (2001). POSTRACK: a low cost real-time motion tracking system for VR application. In *Proceedings seventh international conference on virtual systems and multimedia* (pp. 383–392).

Fernández, I., Lázaro, J. L., Mazo, M., & García, S. (2002). A non linear multicamera system calibration method for tele-operated robot. In *International conference TELEC'02* (Vol. 1, pp. 15–24), Universidad de Alcalá, España.

Fernández, I., Lázaro, J. L., Mazo, M., & García, S. (2003a). Local positioning system (LPS) for indoor environments using a camera array. In *The 11th international conference on advanced robotics. ICAR'2003* (Vol. 2, pp. 613–618), Coimbra, Portugal.

Fernández, I., Lázaro, J. L., Mazo, M., & García, S. (2003b). Sistema de posicionamiento local (LPS) en interiores utilizando un array de cámaras. In *Seminario anual de automática electrónica industrial e instrumentación. SAAEI'2003* (Vol. 1, pp. 165–170), Vigo, España.

Fiala, M. (2004). *Pseudo-random linear image marker (PLIM) self-identifying marker system*. (Tech. Rep. NRC/ERB-1112, NRC 47173). National Research Coucil, Canada, http://iit-iti.nrc-cnrc.gc.ca/iit-publications-iti/docs/NRC-47173.pdf.

Hada, Y., Takase, K., Ohgki, K., Matsukuma, K., Hirata, R., & Okumura, S. (2002). Indor navigation of multiple móbile robots in a dynamic environment using iGPS. In *Proceedings of the 2002 IEEE international conference on robotics & automotion* (pp. 2682–2688), Washington, DC.

Heikkilä, J. (2000). Geometric camera calibration using circular control points. *IEEE Transactions Pattern Analysis and Machine Intelligence*, *25*, 1066–1076.

Hoff, W. A., Nguyen, K., & Lyon, T. (1996). Computer vision-based registration techniques for augmented reality. In *Proceedings of intelligent robots and control systems XV, intelligent control systems and advanced manufacturing* (pp. 538–548).

Hoover, A., & Olsen, B. D. (2000). Sensor network perception for mobile. In *IEEE conference on robotics and automation* (pp. 965–980).

Huabin, T., Lei, W., & Zengqi, S. (2004). Accurate and stable vision in robot soccer. In *Control, automation, robotics and vision conference (ICARCV 2004)* (Vol. 3, pp. 2314–2319).

Kruse, E., & Wahl, F. M. (1998). Camera-based observation of obstacle motions to derive Statistical data for mobile robot motion planning. In *ICRA* (pp. 662–667).

Lavest, J. M., Viala, M., & Dhome, M. (1998) Do we really need an accurate calibration pattern to achieve a reliable camera calibration?. In *ECCV'1998* (Vol. 1, pp. 158–174), Freiburg, Germany.

Lázaro, J. L. (1998). *Modelado de entornos mediante infrarrojos. Aplicación al guiado de robots móviles*. Tesis Doctoral. Escuela Politécnica. Universidad de Alcalá.

Lee, J., Ando, N., Yakushi, T., & Nakajima, K. (2001).Adaptive guidance for mobile robots in intelligent infrastructure. In *Proceedings 2001 IEEE/RSJ interrnational conference on robots and systems* (pp. 90–95).

Liao, P. S., Chen, T. S., & Chung, P. C. (1999). A fast algorithm for multilevel thresholding. *Journal of Information Science and Engineering*, 713–723.

Lilienthal, A., & Duckett, T. (2003). An absolute positioning system for 100 euros. In *Proceedings of the IEEE international workshop on robotic sensing (ROSE 2003)*. ISBN: 0-7803-8109-2.

López, M. E., Bergasa, L. M., Barea, R., & Escudero, M. S. (2005). A navigation system for assistant robots using visually augmented POMDPs. *Autonomous Robots*, *19*, 67–87.

McNamee, L. P. (2003). *Photogrametric calibration of mobile robot kinematics*. Doctoral dissertation, Ottawa-Carleton Institute for Electrical and Computer Engineering. School of Information Technology and Engineering (Electrical and Computer Engineering).

Melen, T. (1994). *Geometrical modelling and calibration of video cameras for underwater navigation*. Doctoral dissertation, Norwegian University of Science and Technology. Trondhein, Norway.

Metronor. (Online). Available: http://www.metronor.com/.

Morioka, K., Mao, X., & Hashimoto, H. (2006). Global color model based object matching in the multi-camera environment. In *2006 IEEE/RSJ international conference on intelligent robots and systems* (pp. 2644–2649).

Naimark, L., & Foxlin, E. (2002). Circular data matrix fiducial system and robust image processing for a wearable vision-interial self-tracker. In *ISMAR'02, international symposium on mixed and augmented reality* (pp. 27–36).

Ocaña, M., Bergasa, L. M., Sotelo, M.Á., Nuevo, J., & Flores, R. (2005). Indoor robot localization system using WIFI signal measure and minimizing calibration effort. In *Proceedings of the ISIE 2005* (pp. 1545–1550), Dubrovnik, Croatia.

Pizarro, D., Santiso, E., & Mazo, M. (2005). Simultaneous localization and structure reconstruction of mobile robots with external cameras. In *ISIE05* (Vol. 3, pp. 1321–1326), Dubrovnik, Croatia.

Santiso, E. (2003). *Posicionamiento absoluto de un robot móvil a partir del reconocimiento de marcas*. Tesis Doctoral. Escuela Politécnica. Universidad de Alcalá.

Sogo, T., Ishiguro, H., & Ishida, T. (1999). Acquisition of qualitative spatial representation by visual observation. In *Proceedings IJCAI* (pp. 1054–1060).

Steinhaus, P., Ehrenmann, M., & Dillmann, R. (1999). MEPHISTO: a modular and extensible path planning system using observation. In *ICVS'99, international conference on computer vision systems* (Vol. 1542, pp. 361–375), Las Palmas, Spain, 13 January 1999. ISBN 3-540-65459.

Steinhaus, P., Walther, M., Giesler, B., & Dillmann, R. (2004). 3D global and mobile sensor data fusion for mobile platform navigation. In *Proceedings of the IEEE international conference on robotics and automation (ICRA '04)* (Vol. 4, pp. 3325–3330).

Takase, K., Hada, Y., & Jia, S. (2003). Autonomous mobile robot platform supported by intelligent information infrastructure. In *Proceedings of the 2003 IEEE international conference on robotics, intelligent systems and signal processing* (pp. 495–500), Changsha, China, October 2003.

Terado, T., & Fujiwara, O. (2004). Measurement of the standing-up motion from a chair using a luminous marker system. In *2004 IEEE international workshop on biomedical circuits and system S2/1-9-12a*.

ViconPeak. (Online). Available: http://www.vicon.com/.

Villadangos, J. M., Ureña, J., Mazo, M., Hernández, A., Martín, M., & Marziani, C. M. (2004). Global positioning of mobile robots by codified ultrasonic beacons. In *Conferencia internacional en telecomunicación, electrónica y control (TELEC-04)* (pp. 47–53), Santiago de Cuba, Cuba.

Welch, G., & Bishop, G. (2001). An introduction to the Kalman filter. In *SIGGRAPH 2001*. Department of Computer Science, University of North Carolina at Chapel Hill.

Yun, J., Park, J., Choi, H., & Lee, J. (2004). Absolute positioning system for mobile robot navigation in an indoor environment. In *ICCAS2004* (pp. 1448–1451), Bangkok, Thailand, 25–27 August 2004.

Ignacio Fernández received a Ph.D. degree from Alcalá University in 2005. He is currently a Professor in the Electronics Department of Alcala University. His research interests are focused to Intelligent Spaces.

Manuel Mazo received a Ph.D. degree in telecommunications in 1988 from the Polytechnic University of Madrid. He is currently a Professor with the Electronics Department, University of Alcalá. His research interests include electronics control, intelligent sensors robot sensing and perception, intelligent spaces.

José L. Lázaro received a Ph.D. degree in telecommunications in 1998. He is currently a Professor in the Electronics Department of Alcala University. His research interests include multisensorial systems applied to the mobile robots.

Daniel Pizarro received the BS and Master degree in electrical and electronical engineering in 2003 from Alcala University. He is currently a PhD student at Department of Electronics in the same university. His research interests are focused on Intelligent Spaces.

Enrique Santiso received the Ph.D. degree in telecommunications from the University of Alcalá. He joined the Department of Electronics, University of Alcalá. His interests are in the fields of Intelligent Spaces.

Pedro Martín received a Ph.D. degree in Telecommunications from Alcala University in 2000. He is currently a Professor in the Electronics Department of Alcala University. His research interests include mainly embedded systems and codesign.

Cristina Losada received the BS and Master degree in electrical and electronical engineering in 2004 from Alcala University. He is currently a PhD student at Department of Electronics in the same university. His research interests are focused on computer vision.