

3D Motion Segmentation and 3D Localization of Mobile Robots Using an Array of Static Cameras and Objective Function Minimization

C. Losada, M. Mazo, S. Palazuelos, D. Pizarro, M. Marrón, F. Redondo

Departamento de Electrónica.
Universidad de Alcalá.
losada@depeca.uah.es

Abstract- This paper presents a method for obtaining the 3D motion segmentation and 3D localization of mobile robots using an array of calibrated cameras that are placed in fixed positions within the environment. This proposal does not rely on previous knowledge or invasive landmarks on board the robots.

The proposal is based on the minimization of an objective function. This function includes information of all the cameras and depends on three groups of variables: the segmentation boundaries, the 3D rigid motion parameters (linear and angular velocity components) and depth (distance to the cameras). For the objective function minimization, we use a greedy algorithm that, after initialization, consists of three iterative steps.

The use of multiple cameras increases notably the system robustness against occlusions and lighting changes. The accuracy of the results is also improved with regard to the methods where a single camera is used.

I. INTRODUCTION

A common problem in the field of autonomous robot guidance is to obtain the position and orientation of the robots within the environment with sufficient accuracy. Several methods have been developed to carry out this task.

The localization methods can be classified into two groups: those that require sensors on board the robots and those that incorporate sensors within the work environment. The second alternative allows to reduce the complexity of the robots. It also facilitates simultaneous navigation of multiple robots within the same environment. This alternative includes “intelligent environments” [1][2] characterized by the use of an array of sensors located in fixed positions of the environment. These sensors are distributed strategically to cover the entire field of movement of the robot. The information provided by the sensors should allow to locate the robots and other mobile objects accurately.

There are several methods to guide mobile robots using an external camera array. The most significant approaches can be divided into two groups. The criterion is based on the previous knowledge about the robot that is required by the method. The first group includes those works that make use

of strong prior knowledge by using artificial landmarks attached to the robot [3]. The second group includes the works that are capable of using the robot natural appearance and the camera geometry to obtain the position [4].

The proposal presented in this paper is included in the second group. It obtains the robot position and orientation using a set of calibrated cameras that are placed in fixed positions within the environment. This proposal does not rely on previous knowledge or invasive landmarks. Robot segmentation and position are obtained through the minimization of an objective function. There are many works which use an objective function [5][6]. However, these works present several disadvantages such as high computational cost or reliance on the initial values of the variables. Moreover, these methods are not robust because they use information from a single camera.

This work has been structured as follows: section II presents the proposed system for 3D motion segmentation and 3D localization of mobile robots. This is made by the minimization of an objective function. Section III describes the objective function. The objective function minimization is described in section IV. Section V shows several experimental results and, finally, section VI presents the main conclusions.

II. PROPOSED SYSTEM

This paper proposes a system that allows the motion segmentation and 3D localization of mobile robots, based only on the images acquired by an array of n_c calibrated and synchronized cameras located in fixed positions within the environment (intelligent space).

Using the work in [5] as a starting point, the motion segmentation and 3D localization are obtained through an objective function minimization. Before the minimization, it is necessary to initialize the variables. Finally, after the function minimization, robot trajectory can be planned. A general block diagram of the proposed system is shown in Fig. 1.

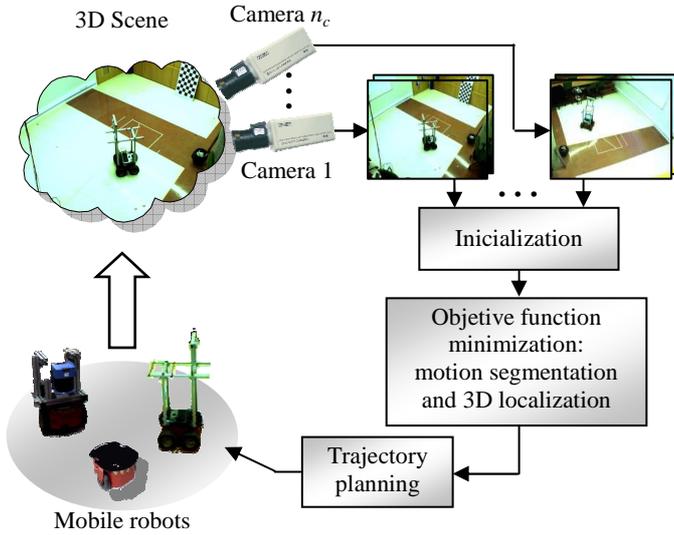


Fig. 1. Block diagram of the proposed system for mobile robot motion segmentation and 3D localization in an intelligent space.

III. OBJECTIVE FUNCTION FOR n_c CAMERAS

To start with, the different coordinate systems used in this work are defined. The 3D coordinates of a point $\mathbf{P} = (X, Y, Z)^T$ can be expressed in the different reference systems of the intelligent space. There is a global reference system named “world coordinate system” and represented by Γ_w . There is also a local reference system associated with each camera (Γ_{c_i} , $i=1, \dots, n_c$) whose origin is located in the center of projection. In Fig. 2, world coordinate system (Γ_w) has been represented in red color and the coordinate systems associated to the cameras (Γ_{c_i}) have been represented in blue color.

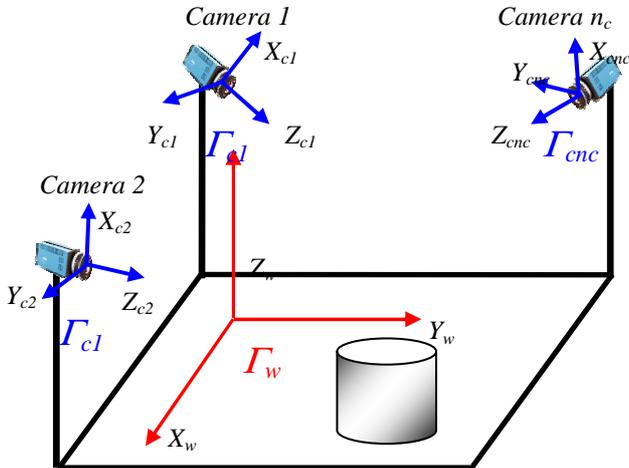


Fig. 2. Reference coordinate systems in the intelligent space: World coordinate system (Γ_w) in red color. Camera coordinate system (Γ_{c_i} $i=1, 2, \dots, n_c$) in blue color.

In this work, the perspective camera model is used. This corresponds to an ideal *pinhole* lens. Perspective projection equations are given by:

$$x = f_u \frac{X_c}{Z_c}, \quad y = f_v \frac{Y_c}{Z_c} \quad (1)$$

where f_u and f_v are the focal lengths of the lens for x and y axis respectively.

A. 3D Brightness Constraint for n_c cameras

This paper presents a generalization of the 3D brightness constraint for rigid objects presented by Sekkati and Mitiche in [5]. The 3D brightness constraint proposed in this work includes the information provided by the n_c cameras available in a intelligent space.

In this section, we do not use i subscript to indicate the camera because the following process is the same for each camera.

Let $\mathbf{P}_w = (X_w, Y_w, Z_w)^T$ be the 3D coordinates of point \mathbf{P} on a mobile robot relative to Γ_w . Let $\mathbf{v}_w = (v_w^x, v_w^y, v_w^z)^T$ and $\boldsymbol{\omega}_w = (\omega_w^x, \omega_w^y, \omega_w^z)^T$ be, respectively, the components of the linear and angular velocity of the robot motion in Γ_w . Then, the velocity of \mathbf{P} , relative to Γ_w , is given by equation (2). In the same way, if $\mathbf{P}_c = (X_c, Y_c, Z_c)^T$ are the coordinates of \mathbf{P} relative to Γ_c and $\mathbf{v}_c = (v_c^x, v_c^y, v_c^z)^T$ and $\boldsymbol{\omega}_c = (\omega_c^x, \omega_c^y, \omega_c^z)^T$ are the components of the linear and angular velocity of the robot motion in Γ_c , the velocity of \mathbf{P} relative to Γ_c is given by equation:

$$\dot{\mathbf{P}}_w = (\dot{X}_w, \dot{Y}_w, \dot{Z}_w)^T = \mathbf{v}_w + \boldsymbol{\omega}_w \times \mathbf{P}_w \quad (2)$$

$$\dot{\mathbf{P}}_c = (\dot{X}_c, \dot{Y}_c, \dot{Z}_c)^T = \mathbf{v}_c + \boldsymbol{\omega}_c \times \mathbf{P}_c \quad (3)$$

Let \mathbf{R}_{wc} be the (3x3) matrix and \mathbf{T}_{wc} the (1x3) translation vector which represent the coordinate transformation from the world coordinate system (Γ_w) to the camera coordinate system (Γ_c). The coordinate transformation is carried out using the expression in equation (4).

$$\mathbf{P}_c = \mathbf{R}_{wc} \mathbf{P}_w + \mathbf{T}_{wc} \quad (4)$$

Deriving the equation (4) with respect to time, and substituting the expressions of the velocities $\dot{\mathbf{P}}_w$ (equation (2)) and $\dot{\mathbf{P}}_c$ (equation (3)) afterwards, the following equation is obtained.

$$\mathbf{v}_c + \boldsymbol{\omega}_c \times \mathbf{P}_c = \mathbf{R}_{wc} (\mathbf{v}_w + \boldsymbol{\omega}_w \times \mathbf{P}_w) \quad (5)$$

Cross product $\boldsymbol{\omega} \times \mathbf{P}$ can be expressed as a scalar product $\hat{\boldsymbol{\omega}} \cdot \mathbf{P}$, where $\hat{\boldsymbol{\omega}}$ is the following matrix:

$$\hat{\boldsymbol{\omega}} = \begin{pmatrix} 0 & -\omega^z & \omega^y \\ \omega^z & 0 & -\omega^x \\ -\omega^y & \omega^x & 0 \end{pmatrix}$$

Using the previous expression, equation (5) can be rewritten to obtain equation (6), where components of linear and angular velocity in Γ_c (\mathbf{v}_c , $\boldsymbol{\omega}_c$) can be expressed in terms of the components of velocity in Γ_w (\mathbf{v}_w , $\boldsymbol{\omega}_w$) and the transformation matrices (\mathbf{R}_{wc} , \mathbf{T}_{wc}).

$$\mathbf{v}_c = \mathbf{R}_{wc} \mathbf{v}_w - \mathbf{R}_{wc} \hat{\boldsymbol{\omega}}_w \mathbf{R}_{wc}^T \mathbf{T}_{wc} \\ \boldsymbol{\omega}_c = \text{adj}(\mathbf{R}_{wc}) \boldsymbol{\omega}_w \quad (6)$$

If (x, y) are the coordinates of the projection on the image plane of a point \mathbf{P} , the derivative of the perspective projection equations (1) with respect to time, and the subsequent substitution of the expression of the velocity components of \mathbf{P} in Γ_c (3) allow us to obtain the following equations for the image motion components (\dot{x}, \dot{y}) :

$$\dot{x} = \frac{1}{Z_c} (f_u R_{wc}^1 - x R_{wc}^3) \mathbf{v}_w + \mathbf{q}_u \text{adj}(\mathbf{R}_{wc}) \boldsymbol{\omega}_w \quad (7)$$

$$\dot{y} = \frac{1}{Z_c} (f_v R_{wc}^2 - y R_{wc}^3) \mathbf{v}_w + \mathbf{q}_v \text{adj}(\mathbf{R}_{wc}) \boldsymbol{\omega}_w \quad (8)$$

where R_{wc}^i is the i^{th} row in the rotation matrix from Γ_w to Γ_c (\mathbf{R}_{wc}) and $\mathbf{q}_u, \mathbf{q}_v$ are the following (1x3) vectors:

$$\mathbf{q}_u = \begin{pmatrix} x \begin{bmatrix} \frac{f_u}{Z_c} - \frac{y}{f_u} \\ \frac{f_u}{Z_c} - \frac{y}{f_u} \end{bmatrix} \\ \left[f_u + \frac{x^2}{f_u} - \frac{1}{Z_c} (f_u t_{wc}^z + x t_{wc}^x) \right] \\ f_u \begin{bmatrix} \frac{f_u}{Z_c} - \frac{y}{f_u} \\ \frac{f_u}{Z_c} - \frac{y}{f_u} \end{bmatrix} \end{pmatrix}^T \quad (9)$$

$$\mathbf{q}_v = - \begin{pmatrix} \left[f_v + \frac{y^2}{f_v} - \frac{1}{Z_c} (f_v t_{wc}^z + y t_{wc}^y) \right] \\ y \begin{bmatrix} \frac{f_v}{Z_c} - \frac{x}{f_v} \\ \frac{f_v}{Z_c} - \frac{x}{f_v} \end{bmatrix} \\ f_v \begin{bmatrix} \frac{f_v}{Z_c} - \frac{x}{f_v} \\ \frac{f_v}{Z_c} - \frac{x}{f_v} \end{bmatrix} \end{pmatrix}^T \quad (10)$$

Substitution of velocity components in the image plane (denoted as u and v) in brightness constraint (11) allows to obtain a 3D brightness constraint for rigid objects in terms of the linear and angular velocity components in Γ_w (\mathbf{v}_w and $\boldsymbol{\omega}_w$). This 3D brightness constraint is shown in equation (12).

$$I_x u + I_y v + I_t = 0 \quad (11)$$

$$\Psi_k(\mathbf{x}) = I_t + \mathbf{s} \cdot \mathbf{R}_{wc} \frac{\mathbf{v}_w}{Z_c} + \mathbf{q} \cdot \text{adj}(\mathbf{R}_{wc}) \boldsymbol{\omega}_w + \quad (12)$$

$$T_{wc}^T \mathbf{r} \cdot \text{adj}(\mathbf{R}_{wc}) \frac{\boldsymbol{\omega}_w}{Z_c} = 0$$

In equation (12), \mathbf{s} and \mathbf{q} are two vectors with dimension (1x3), and \mathbf{r} is a (3x3) matrix. These matrices are given, respectively, by equations (13), (14) and (15):

$$\mathbf{s} = [f_u I_x \quad f_v I_y \quad -(x I_x + y I_y)] \quad (13)$$

$$\mathbf{q} = \begin{pmatrix} -f_v I_y - \frac{y}{f_v} (x \cdot I_x + y \cdot I_y) \\ + f_u I_x + \frac{x}{f_u} (x \cdot I_x + y \cdot I_y) \\ -\frac{f_u}{f_v} y \cdot I_x + \frac{f_v}{f_u} x \cdot I_y \end{pmatrix}^T \quad (14)$$

$$\mathbf{r} = \begin{pmatrix} 0 & -(x I_x + y I_y) & -f_v I_y \\ (x I_x + y I_y) & 0 & f_u I_x \\ f_v I_y & -f_u I_x & 0 \end{pmatrix} \quad (15)$$

3D brightness constraint shown in equation (12) must be satisfied in all of the n_c cameras. Knowing it, we define a new 3D brightness constraint for rigid objects which includes all the information provided by the n_c cameras available in the intelligent space (equation (16)). Constraint in equation (16) is defined for each region, in each camera. If there are $N-1$ robots in a scene, the scene is divided into N regions (region N corresponds to the background). Then, we add two subscripts to denote a region: subscript k ($k=1,2,\dots,N$), which indicates the region, and subscript i ($i=1,2,\dots,n_c$) which indicates the camera.

$$\Psi_{ki}(\mathbf{x}) = I_{ti} + \mathbf{s}_i \cdot \mathbf{R}_{wci} \cdot \frac{\mathbf{v}_{wk}}{Z_{ci}} + \quad (16)$$

$$\mathbf{q}_i \cdot \text{adj}(\mathbf{R}_{wci}) \boldsymbol{\omega}_{wk} + \mathbf{t}_{wci}^T \mathbf{r}_i \cdot \text{adj}(\mathbf{R}_{wci}) \frac{\boldsymbol{\omega}_{wk}}{Z_{ci}}$$

3D brightness constraint in equation (16) can be written more compactly as shown in equation (17):

$$\Psi_{ki}(\mathbf{x}) = I_{ti} + \mathbf{S}_i \frac{\mathbf{v}_{wk}}{Z_{ci}} + \left(\mathbf{Q}_i + \frac{\mathbf{R}_i}{Z_{ci}} \right) \boldsymbol{\omega}_{wk} \quad (17)$$

where:

$$\mathbf{S}_i = \mathbf{s}_i \cdot \mathbf{R}_{wci}, \quad \mathbf{Q}_i = \mathbf{q}_i \cdot \text{adj}(\mathbf{R}_{wci}) \quad \text{y} \quad \mathbf{R}_i = \mathbf{t}_{wci}^T \cdot \mathbf{r}_i \cdot \text{adj}(\mathbf{R}_{wci})$$

In equation (17), it is necessary to take into account that 3D velocity components \mathbf{v}_{wk} and $\boldsymbol{\omega}_{wk}$ are equal for the n_c cameras, whereas I_{ti} depends on the measurements in each camera and the matrices \mathbf{S}_i , \mathbf{Q}_i y \mathbf{R}_i depend on the intrinsic parameters and transformation matrices (\mathbf{R}_{wci} and \mathbf{T}_{wci}) of each camera.

B. Objective Function for n_c Cameras

The objective function for n_c cameras proposed in this work (equation (18)) depends on three groups of variables.

1. A set of $N-1$ curves which defines the $N-1$ mobile robot segmentation boundaries in the images acquired by each camera $\{\gamma_{ki}\}_{k=1,\dots,N-1}^{i=1,\dots,n_c}$. These curves divide each image in N regions $\{\Omega_{ki}\}_{k=1,\dots,N-1}^{i=1,\dots,n_c}$ (where Ω_{ki} represents the set of points of the image i inside the curve k γ_{ki}).
2. The components of linear and angular velocity of ($N-1$) mobile robots and background, related to Γ_w : $\{\mathbf{v}_{wk}\}_{k=1}^N$, $\{\boldsymbol{\omega}_{wk}\}_{k=1}^N$. (These velocities are the same from all cameras).
3. The depth (distance from each 3D point (\mathbf{P}) to each camera) which is the Z_{ci} coordinate of point \mathbf{P} (related to Γ_{ci}).

$$E \left[\{\gamma_{ki}\}_{k=1,\dots,N-1}^{i=1,\dots,n_c}, \{\mathbf{T}_{wk}\}_{k=1}^N, \{\boldsymbol{\omega}_{wk}\}_{k=1}^N, \{Z_{ci}\}_{i=1}^{n_c} \right] = \sum_{k=1}^N \sum_{i=1}^{n_c} \left[\int_{\Omega_{ki}} \Psi_{ki}^2(\mathbf{x}) d\mathbf{x} + \mu \int_{\Omega_{ki}} g(\|\nabla Z_{ci}\|) d\mathbf{x} \right] + \sum_{k=1}^{N-1} \sum_{i=1}^{n_c} \lambda \oint_{\gamma_{ki}} ds \quad (18)$$

$$\lambda, \mu \in \mathfrak{R}, \lambda, \mu > 0$$

In equation (18), Ψ_{ki} is the 3D brightness constraint (defined in equation (16)) for the pixels inside the curve k in image i (γ_{ki}); λ y μ are positive, real constants to weigh the contribution of the terms in (18) and $\nabla = (\partial_x, \partial_y)$ is the spatial gradient operator.

The first term in (18) measures conformity of 3D interpretation to the sequence spatiotemporal variations in each region (Ω_{ki}) through the 3D brightness constraint. The second integral is a regularization term of smoothness of depth, and the third integral is a regularization term of the $N-1$ boundaries.

IV. OBJECTIVE FUNCTION MINIMIZATION FOR n_c CAMERAS

Because the objective function (defined in equation (18)) depends on three groups of variables, a greedy algorithm, which consists of three iterated steps, is used. After curve and depth initialization (in each Γ_{ci} , $i=1,\dots,n_c$), the three steps are repeated until the algorithm converges.

A. Initialization

There are several alternatives for curve initialization. In this work, since cameras are located in fixed positions within the environment, the $N-1$ initial curves are obtained using

GPCA (Generalized Principal Components Analysis) [7]. GPCA technique is used in this work to obtain a background model. GPCA is also used to compare each image to the background model [8][9].

The initial depth relative to each camera coordinate system (Γ_{ci}) is obtained using Visual Hull 3D [10] which allows to obtain a 3D occupancy grid (composed of cubes whose size is Δh) in Γ_w . The 3D occupancy grid is obtained from the initial segmentation boundaries in each image, which are computed using GPCA.

The 3D coordinates of the occupied cell are projected from Γ_w to each camera coordinate system Γ_{ci} ($i=1, \dots, n_c$) through the transformation matrices (\mathbf{R}_{wci} and \mathbf{T}_{wci}) to obtain a set of points on the mobile robot in Γ_{ci} . This provides an effective method for depth initialization in each camera.

In Fig. 3, a block diagram including stages involved in curves and depth initialization for n_c cameras is shown.

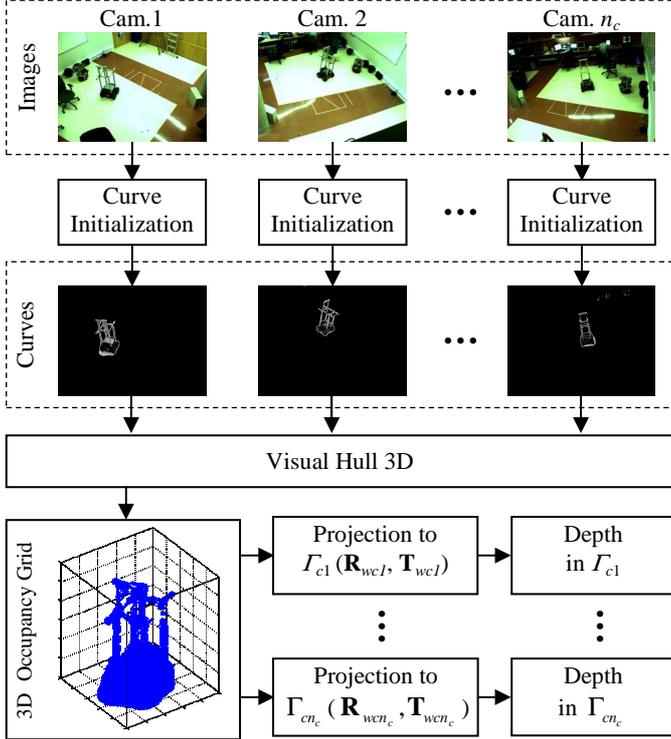


Fig. 3. Block diagram of the proposed algorithm for curve and depth initialization using GPCA and Visual Hull 3D.

B. 3D motion by linear least squares

At this step, segmentation boundaries and depth in each Γ_{ci} are fixed. So, the energy to minimize reduces to:

$$E(\{\mathbf{v}_{wk}\}_{k=1}^N, \{\boldsymbol{\omega}_{wk}\}_{k=1}^N) = \sum_{k=1}^N \sum_{i=1}^{n_c} \int_{\Omega_{ki}} \psi_{ki}^2(\mathbf{x}) d\mathbf{x} \quad (19)$$

Taking into account that the 3D brightness constraint for n_c cameras (equation (17)) depends linearly on \mathbf{v}_{wk} and $\boldsymbol{\omega}_{wk}$, 3D motion parameters may be obtained by linear least squares.

Let p_{ki} be the number of the image acquired by the i^{th} camera, within region k ($k=1, \dots, N$), and let $\mathbf{a}_{ki}(\mathbf{x}_j)$ be the 1×6 vector associated to each point \mathbf{x}_j ($j = 1, 2, \dots, p_{ki}$) in the i^{th} image, them:

$$\mathbf{a}_k(\mathbf{x}_j) = \left(\frac{S_{i1}}{Z_{ci}}, \frac{S_{i2}}{Z_{ci}}, \frac{S_{i3}}{Z_{ci}}, \mathbf{Q}_{i1} + \frac{\mathbf{R}_{i1}}{Z_{ci}}, \mathbf{Q}_{i2} + \frac{\mathbf{R}_{i2}}{Z_{ci}}, \mathbf{Q}_{i3} + \frac{\mathbf{R}_{i3}}{Z_{ci}} \right) \quad (20)$$

Let $\boldsymbol{\rho}_{wk} = (\mathbf{v}_{wk}, \boldsymbol{\omega}_{wk})^T$ be the 1×6 vector representing the six-dimensional rigid motion components of the robot associated to region Ω_{ki} in Γ_w reference system. \mathbf{A}_k and \mathbf{c}_k are defined as:

$$\mathbf{A}_k = \left(\mathbf{a}_{k1}(x_{11}), \dots, \mathbf{a}_{k1}(x_{p_{k1}}), \mathbf{a}_{k2}(x_{12}), \dots, \mathbf{a}_{kn_c}(x_{p_{kn_c}}) \right)^T$$

$$\mathbf{c}_k = \left(-I_{i1}(x_{11}), \dots, -I_{i1}(x_{p_{k1}}), -I_{i2}(x_{12}), \dots, -I_{in_c}(x_{p_{kn_c}}) \right)^T$$

$k=1, \dots, N$

Linear \mathbf{v}_{wk} and angular $\boldsymbol{\omega}_{wk}$ velocity components can be obtained solving the linear equation system shown in (21).

$$\mathbf{A}_k \boldsymbol{\rho}_{wk} = \mathbf{b}_k \quad k=1, \dots, N \quad (21)$$

In case mobile robot movement is restricted to XY plane, equation (21) can be simplified because linear velocity along the Z_w axis and also angular velocity along X_w and Y_w axes are null. Then, both vectors $\boldsymbol{\rho}_{wk}$ and $\mathbf{a}_{ki}(\mathbf{x}_j)$ consist only of three components.

C. Depth estimation by gradient descent

In the second step, the function to minimize in order to recover the depth is shown in equation (22). In this function, χ_{ki} is the characteristic function of region k in image i (Ω_{ki}).

$$E(Z) = \sum_{k=1}^N \sum_{i=1}^{n_c} \int_{\Omega_{ki}} [\psi_{ki}^2(\mathbf{x}) + \mu g(\|\nabla Z_{ci}\|)] d\mathbf{x} \quad (22)$$

$$= \int_{\Omega} \sum_{k=1}^N \sum_{i=1}^{n_c} [\chi_{ki}(\mathbf{x}) (\psi_{ki}^2(\mathbf{x}) + \mu g(\|\nabla Z_{ci}\|))] d\mathbf{x}$$

Given a set of curves $\{\gamma_{ki}\}_{k=1, \dots, N-1}^{i=1, \dots, n_c}$ that divides the image acquired by each camera in N regions $\{\Omega_{ki}\}_{k=1}^N$, the descent equations for any region and for any camera are shown in equation (23). In equation (23), τ indicates the algorithm execution time and g' is the ordinary derivative of the boundary preserving function g .

$$\frac{\partial Z_{ci}}{\partial \tau} = \frac{2}{Z_{ci}^2} (\mathbf{S}_i \mathbf{v}_{wk} + \mathbf{R}_i \boldsymbol{\omega}_{wk}) \psi_{ki} + \mu \text{div} \left(\frac{g'(\|\nabla Z_{ci}\|)}{\|\nabla Z_{ci}\|} \nabla Z_{ci} \right) \quad (23)$$

$i=1, \dots, n_c, \quad k=1, \dots, N$

In this work we use a quadratic boundary preserving function ($g(a) = a^2$). Boundary preserving function efficacy has been verified in several experiments. In these experiments we have obtained similar results using the quadratic function, and more complex functions proposed in the work of Aubert et al. [11].

D. Curve Evolution by level sets for 3D motion segmentation

With depth Z_{ci} and 3D motion parameters $\{\mathbf{v}_{wk}, \boldsymbol{\omega}_{wk}\}_{k=1}^N$ fixed, the energy to minimize respect to the curves that define the mobile robot contours $\{\gamma_{ki}\}_{k=1, \dots, N-1}^{i=1, \dots, n_c}$ in each image is shown in equation (24), where $\xi_{ki}(\mathbf{x}) = \psi_{ki}^2(\mathbf{x}) + \mu g(\|\nabla Z_{ci}\|)$.

$$E[\{\gamma_{ki}\}_{k=1, \dots, N-1}^{i=1, \dots, n_c}] = \sum_{k=1}^N \sum_{i=1}^{n_c} \int_{\Omega_{ki}} \xi_{ki}(\mathbf{x}) d\mathbf{x} + \lambda \sum_{k=1}^{N-1} \sum_{i=1}^{n_c} \oint_{\gamma_{ki}} ds \quad (24)$$

For multiple region segmentation, the following Euler-Lagrange descent equations are obtained [5]:

$$\frac{\partial \gamma_{ki}}{\partial \tau}(\tau) = -(\xi_{ki}(\gamma_{ki}) - \varphi_{ki}(\gamma_{ki}) + \lambda \kappa_{\gamma_{ki}}(\gamma_{ki})) \times \mathbf{n}_{ki}(\gamma_{ki}) \quad (25)$$

$$i = 1, \dots, n_c \quad k = 1, \dots, N-1$$

where $\kappa_{\gamma_{ki}}$ is the mean curvature of contour defined by γ_{ki} , \mathbf{n}_{ki} is the exterior unit normal function to the curve γ_{ki} and functions φ_{ki} are defined by equation (26).

$$\varphi_{ki}(\gamma_{ki}(s)) = \min_{j \neq k} \xi_{ji}(\gamma_{ki}(s)) \quad (26)$$

After initialization, the three steps described in this section are repeated until the algorithm convergence. The algorithm converges when the computed variables cease to evolve significantly.

V. EXPERIMENTAL RESULTS

In order to validate the proposed system, several experiments have been carried out. In these experiments, we have used a one-hundred image sequence containing a mobile robot. These images have been acquired using four calibrated cameras, placed in fixed positions, within the intelligent space. Fig. 4 shows three images belonging to the test sequence. These images have been acquired at the same time, but using three different cameras.



Fig. 4. Images acquired using three of the four available cameras in the intelligent space.

In this section we provide the experimental results obtained for a one-hundred image sequence using the proposed algorithm for motion segmentation and 3D localization with four cameras, and using the objective function defined in [7] for one camera. Motion segmentation and 3D position have been obtained for each couple of images in the sequence.

It is noteworthy that, to compare the experimental results obtained with both algorithms, the initial values of curves and depth used are the same. Moreover, λ and μ values have been set in an experimental way for both algorithms, using a different image sequence that has been acquired using the same calibrated cameras. It is noteworthy that the proposed algorithm for curve and depth initialization increases notably the robustness of the system against the values of the constants.

Objective function minimization allows to obtain both linear and angular components of the mobile robot velocity and the robot 3D position.

Fig. 5 shows linear velocity estimation errors (in millimeters/frame) for each couple of images in the sequence using the objective function minimization with one camera [7] and the proposal in this work with four cameras. This error has been calculated using equation (27) where ϵ_{vx} and ϵ_{vy} are the difference between the velocity estimated using the algorithm and the velocity measured by the odometric sensors on board the mobile robot along X_w and Y_w axis respectively.

$$\epsilon_v = \sqrt{\frac{1}{2}(\epsilon_{vx}^2 + \epsilon_{vy}^2)} \quad (27)$$

In equation (27), we have only included two velocity components because robot's movement is restricted to XY plane (linear velocity along the Z_w axis is always zero).

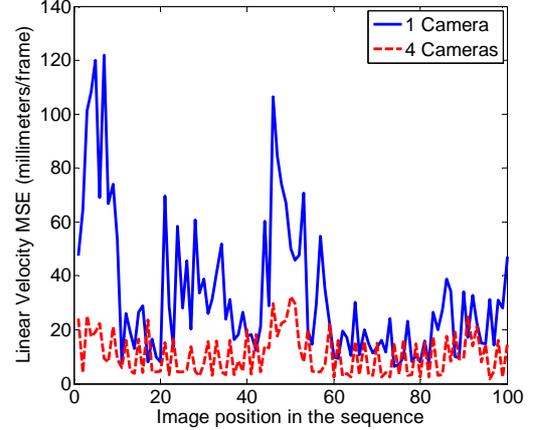


Fig. 5. Linear velocity mean square error (in millimeters/frame) for each couple of images in the sequence using the algorithm for one camera (in blue color) and the proposed algorithm with four cameras (in red color).

Fig. 5 shows that 3D position is obtained more accurately using the proposed algorithm.

Positioning error (in millimeters) is shown in Fig. 6. This error is calculated as the difference between the estimated and the measured position along X_w and Y_w axis.

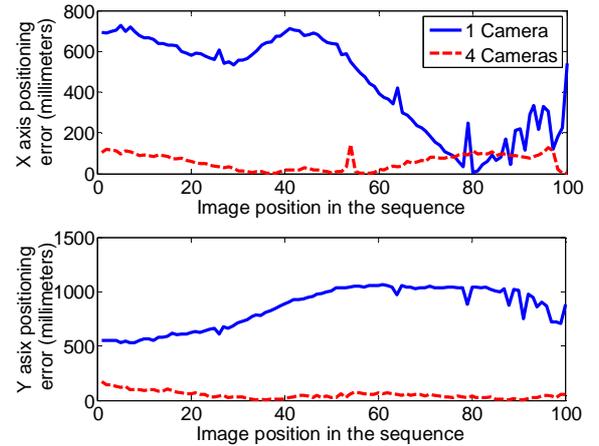


Fig. 6. 3D positioning error (along X and Y axis) obtained for each couple of images in the sequence using the algorithm for one camera (in blue color) and the proposed algorithm with four cameras (in red color).

Fig. 7 shows the position estimated by the algorithm and measured in XY plane.

Positions shown in Fig. 7 have been obtained from a 400 image sequence. Estimated position using the proposed algorithm with four cameras is indicated in discontinuous red line, and measured position is indicated in black line.

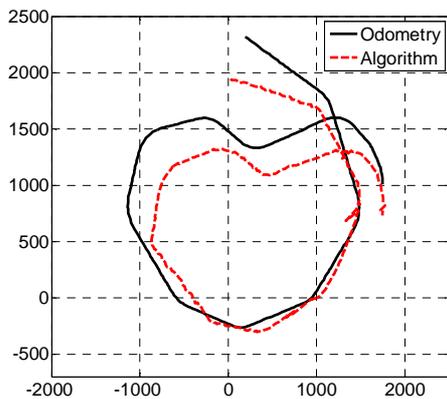


Fig. 7. Estimated 3D position obtained using the proposed algorithm with four cameras (indicated in red color) and measured by robot's odometric sensors (in black color)

The processing times of the algorithms have also been compared. Fig. 8 shows that the processing times of both algorithms are similar, although the proposed algorithm includes information acquired by four cameras, instead of information from only one camera.

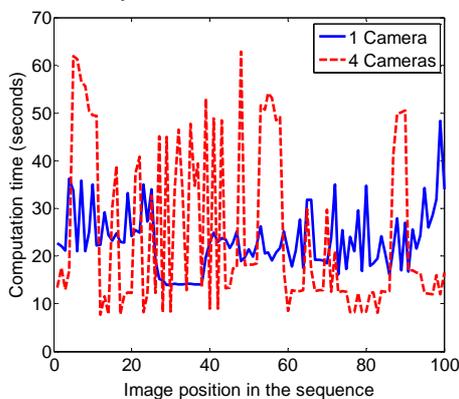


Fig. 8. Processing time (in seconds) for each couple of images in the sequence using the algorithm for one camera (in blue color) and the proposed algorithm with four cameras (in red color).

Finally, in Table I, a summary of the average values of processing time, number of iterations, and position and velocity error obtained for 100 images are shown. In Table I, results obtained using only one camera, and results obtained using the proposed algorithm with four cameras are included.

TABLE I

AVERAGE VALUE OF PROCESSING TIME, NUMBER OF ITERATIONS, AND POSITION AND VELOCITY ERRORS OBTAINED FOR 100 IMAGES

	1 camera	4 cameras
Processing time (seconds)	23.1357	24.9038
Number of iterations	18.55	5.97
Velocity MSE (millimeters/frame)	33.0301	10.8792
Position MSE (millimeters)	706.0428	56.62

Table I shows that processing time is similar for both algorithms, with one or four cameras. However, the proposed algorithm with four cameras increases, notably, the accuracy in the obtained results. Its increase is especially noticeable in the 3D position.

VI. CONCLUSIONS

This paper has presented a system for 3D motion segmentation and 3D positioning of mobile robots using an array of calibrated cameras that are placed in fixed positions within the environment (intelligent space). The proposed algorithm is based on the minimization of an objective function and it is a generalization for n_c cameras of the function proposed by Sekkati and Mitiche in [7].

Visual Hull 3D is used to relate the information from all the available cameras in the world coordinate system Γ_w .

Several experimental tests have been carried out and the obtained results validate the proposal presented in this paper. It has been demonstrated that the proposed algorithm for n_c cameras increases notably the accuracy in motion segmentation and 3D positioning (with respect to the algorithm with only one camera). Moreover, the improvement is obtained without increasing the algorithm processing time.

ACKNOWLEDGMENTS

The work described in this paper was supported by the Ministry of Science and Technology (MEC) under RESELAI project (REF-TIN2006-14896-C02-01).

REFERENCES

- [1] Lee, J., Ando, N., Yakushi, T., & Nakajima, K. Adaptive guidance for Mobile robots in intelligent infrastructure. In Proc. IEEE/RSJ International conference on robots and Systems 2001, pp. 90-95.
- [2] Steinhaus, P., Walther, M., Giesler, B., & Dillmann, R. 3D global and Mobile sensor data fusion for Mobile platform navigation. In Proc. Of the IEEE International conference on robotics and automation (ICRA 2004). Vol. 4, pp. 3325-3330.
- [3] Sogo, T., Ishiguro, H. & Ishida, T. Acquisition of qualitative spatial representation by visual observation. In Proc. IJCAI, pp. 1054-1060. 1999.
- [4] Pizarro, D., Santiso, E & Mazo, M. Simultaneous localization and structure reconstruction of mobile robots with external cameras. In ISIE05, vol. 3, pp. 1321-1326. 2005.
- [5] Sekkati, H. & Mitiche, A. Concurrent 3D Motion Segmentation and 3D Interpretation of Temporal Sequences of Monocular Images. IEEE Transactions on Image Processing, Vol. 15, No. 3, pp. 641-653, 2006.
- [6] Sekkati, H. & Mitiche, A. Joint Optical Flow Estimation, Segmentation, and Interpretation with Level Sets. Computer Vision and Image Understanding, Vol. 103, No. 2, pp. 89-100. 2006.
- [7] Ye, Jieping; Janardan, Ravi; Li, Qi. GPCA: an efficient dimension reduction scheme for image compression and retrieval. Proc. of the 10th ACM SIGKDD international conference on Knowledge discovery and data mining, pp. 354-363. 2004.
- [8] Losada, C., Mazo, M., Palazuelos, S. & Redondo, F. Segmentación robots móviles en espacios inteligentes utilizando técnicas GPCA y minimización de funciones de energía. XV Seminario Anual de Automática, Electrónica Industrial e Instrumentación (SAAEI08). ISBN: 13:978-84-96997-04-2. 2008.
- [9] Losada, C., Mazo, M., Palazuelos, S. & Blanco, E. Segmentación y posicionamiento de sillas de ruedas en espacios inteligentes mediante minimización de funciones de energía. V Congreso de tecnologías de apoyo a la discapacidad (IBERDISCAP 2008). ISBN: 978-958-8316-63-5. pp. 271-274. 2008.
- [10] Laurentini, A. The Visual Hull: a new tool for contour-based image understanding. In Proc. of 7th Scandinavian Conference on Image Processing, pp. 993-1002. 2001.
- [11] G. Aubert, R. Deriche, and P. Kornprobst. Computing optical flow via variational techniques. SIAM Journal on Applied Mathematics 1999. Vol. 60, Issue 1. Pages:156-182.