Indoor localization by using stereoscopic vision, odometry, and the Kalman Filter

Conference Paper · October 2014

DOI: 10.1109/ANDESCON.2014.7098554

2 authors:


Andres Rosales
Escuela Politécnica Nacional


David Fernando Pozo
Universidad de Las Américas

Indoor Localization by Using Stereoscopic Vision, Odometry, and the Kalman Filter

(Preprinted versión. This is not for mass dissemination.)

C. Revelo, M. Trujillo, A. Rosales, *Member, IEEE* and D. Pozo, *Member, IEEE*

***Abstract*— In order to obtain the location of two robotic platforms within a controlled environment in terms of color and brightness, an algorithm based on Kalman Filter has been developed. This algorithm considers the platform's odometry as the process and stereoscopic vision as the external observer. The tracking is implemented through stereoscopic vision and color object detection. Finally, the algorithm was tested by using Robotino® platforms from Festo, showing a good performance even with the loss of the observer.**

***Index Terms*— Tracking, Stereoscopic Vision, Camera´ calibration, 3D point, Kalman Filter.**

I. Introduction

Localization is a fundamental part of Mobile Robotics, since it looks to interpret the physical behavior of an object in an environment [8, 9, 10]. The perception of motion of an object is an easy task for human beings, but the computational implementation of it requires a great deal processing capacity due to the amount of information to be obtained and further processing, resulting errors in the estimation of the true position of the analyzed object in many cases [11, 12].

Modern technology provides several methods which are used for locating objects. One of these is the probabilistic method known as the Monte Carlo method, also known as Particle Filter, where a robot detects another robot in the same environment, thus getting a location system more accurately and quickly than if both robots had done it individually [1].In addition, it is possible to attain the localization based on odometry by estimating the position of the robot platforms during the navigation through the information provided by encoders placed on wheels [11, 12]. This type of localization provides good precision in the short term, but through time error is accumulated, as even odometry is based on the assumption that wheel speed can always be translated to a linear movement relative to the floor [2]. A widely used method at the present time is stereoscopic vision, in which it is possible to reverse the process in order to calculate the 3D position of an object if its position in each image is known, given the information of a pair of stereo cameras. This is permissible since projection lines match exactly at a point in space [3]. The main disadvantages of this method are due to occlusion and lighting conditions, and that to achieve successful results an accurate calibration of cameras is required. One of the methods used today to correct deviations in time is the Kalman filter, especially when it is about processes that can be defined discretely, since this filter predicts the new position of the robotic platforms and is regulated by itself with each new measurement in defined instants of time [4].

Keeping in mind the previously described arguments in relation to the problem of locating and looking for a solution whose results are reliable, an algorithm was implemented in order to locate two robotic Robotino® platforms from Festo [13] in controlled environments using a stereoscopic vision system and odometry, where the platform which has the blue color will be called Robotino1 while the other one which has the red color will be referred to as Robotino2. To obtain the location of two moving objects, the Kalman Filter has been implemented between the coordinates obtained by stereoscopic vision and those obtained by the odometry of the robotic platform. Furthermore, the object segmentation technique based on color characteristics was used in order to identify each of the robotic platforms utilized (see Fig. 1).

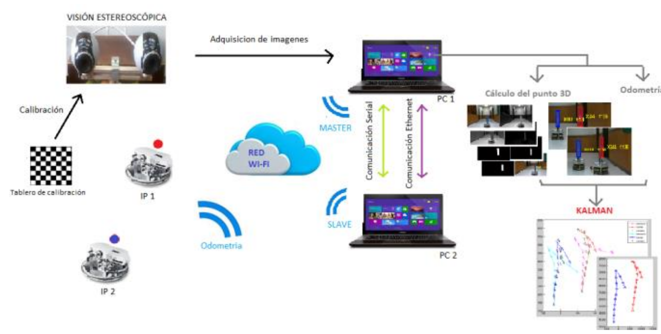


Fig.1. System Operation

II. Obtaining the coordinates of a 3D point from an object using stereoscopic vision

Stereoscopic vision is the attempt to reproduce the behavior of human vision through photographs that are static images representing reality in order to obtain information from the physical world, using a computer.

In order to obtain the 3D coordinates of an object point from a stereoscopic vision system, the following information is required: [3]

1. Find P1and P2, the projection matrix for each camera.
2. Locate the position of the point ($x1$, $y1$) and ($x2$, $y2$) in the image from each camera (x, y).
3. Applying the following equations [3].

$$\begin{bmatrix} x_1p_{31}^1 - p_{11}^1 & x_1p_{32}^1 - p_{12}^1 & x_1p_{33}^1 - p_{13}^1 \\ y_1p_{31}^1 - p_{21}^1 & y_1p_{32}^1 - p_{22}^1 & y_1p_{33}^1 - p_{23}^1 \\ x_2p_{31}^2 - p_{11}^2 & x_2p_{32}^2 - p_{12}^2 & x_2p_{33}^2 - p_{13}^2 \\ y_2p_{31}^2 - p_{21}^2 & y_2p_{32}^2 - p_{22}^2 & y_2p_{33}^2 - p_{23}^2 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} p_{14}^1 - x_1p_{34}^1 \\ p_{24}^1 - y_1p_{34}^1 \\ p_{14}^2 - x_2p_{34}^2 \\ p_{24}^2 - y_2p_{34}^2 \end{bmatrix} \quad (1)$$

$$AX = B \quad (2)$$

$$(A^TA)^{-1}A^TAX = (A^TA)^{-1}A^TB \quad (3)$$

$$X = (A^TA)^{-1}A^TB \quad (4)$$

A. The projection matrix

The projection matrix P is the relationship between the 3D coordinates of an object point and the point inside the image in the 2D image coordinates. It consists of intrinsic and extrinsic camera parameters obtained from the single and stereoscopic calibration [3]:

$$P = \begin{bmatrix} fx & s & cx & 0 \\ 0 & fy & cy & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} R & t \\ 0 & 1 \end{bmatrix} \quad (5)$$

Where:
t=Translation matrix
R=Rotation matrix
fx, fy= Principal point
cx, cy=Pixel size
s= Skew coefficient

The calibration was made by using the Camera Calibration Toolbox for Matlab software. With the obtained results, the projection matrices $P1$ and $P2$ were formed, and they are shown below.

Left Camera: The left camera is considered as the origin of the global coordinate system, where its extrinsic parameters are given by the matrix R, which is equal to the identity matrix, and the translation matrix, which is equal to zero.

Intrinsic parameters of left camera:

Focal Length: fc_left = [642.71017 644.30261] ± [8.63064 9.09298]

Principal point: cc_left = [335.58339 213.87465] ± [10.77779 12.87846]

Skew: alpha_c_left = [0.00000] ± [0.00000] => angle of pixel axes = 90.00000 ± 0.00000 degrees

Distortion: kc_left = [0.07686 -0.17716 -0.00246 0.00091 0.00000] ± [0.03561 0.12929 0.00676 0.00542 0.00000]

$$P1 = \begin{bmatrix} 642.71017 & 0 & 335.58339 & 0 \\ 0 & 644.30261 & 213.87465 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

RightCamera:

Intrinsic parameters of right camera:

Focal Length: fc_right = [642.19825 643.89559] ± [6.55484 6.91735]

Principal point: cc_right = [341.92842 223.33536] ± [7.26314 9.65685]

Skew: alpha_c_right = [0.00000] ± [0.00000] => angle of pixel axes = 90.00000 ± 0.00000 degrees

Distortion: kc_right = [0.08563 -0.24416 -0.00551 0.00340 0.00000] ± [0.02822 0.09484 0.00477 0.00356 0.00000]

Extrinsic parameters (position of right camera wrt left camera):

Rotation vector: om = [0.04138 0.02443 -0.01481]

Translation vector: T = [-104.87624 1.02964 4.87007]

R= [0.9996 0.0153 0.0241; -0.0143 0.9990 -0.0415; -0.0247 0.0412 0.9988]

$$P2 = \begin{bmatrix} 633 & 24 & 357 & 65686 \\ -15 & 652 & 196 & 1751 \\ 0 & 0 & 1 & 5 \end{bmatrix}$$

B. Location of the object in the image

In order to find the values of pixels (x, y) that represent the object, the following scheme (Fig. 2) is applied:

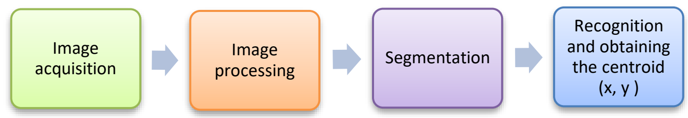


Fig.2. Scheme for obtaining the points (*x1, y1*) and (*x2, y2*) of the objects of interest.

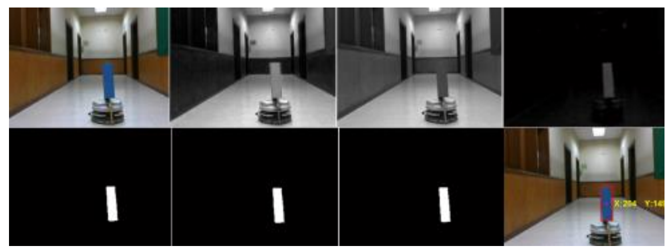


Fig. 3.Recognition process and obtaining blue object centroid

The acquisition image processing consists of two threads: the capture by using an optical scanning device and the digitizing which transforms the obtained information into a digital image ready to be processed.

The segmentation of the object is obtained by a combination of color segmentation, which works with the RGB components of the image, and thresholding segmentation, which binarizes the image within a given value.

Having isolated the object of interest by selecting characteristics of color, the region value that it comprises, and so on, the region was labeled in order to obtain its centroid position coordinates (x, y) of the center pixel of the region (see Fig. 3).

III. Kalman filter

This filter is considered as an optimal estimator which can be implemented through linear or nonlinear ways. It performs a prediction process that can be understood as an update of the time and a step where the measurement is updated.

The predicting and updating equations are shown below:

Predicting Equations

$$\hat{x}_k^- = A\hat{x}_{k-1} + Bu_{k-1} + w_{k-1} \quad (6)$$

$$P_k^- = AP_{k-1}A^T + Q \quad (7)$$

Updating Equations

$$K_k = P_k^-H^T(HP_k^-H^T + R)^{-1} \quad (8)$$

$$\hat{x}_k = \hat{x}_k^- + K_k(z_k - H\hat{x}_k^-) \quad (9)$$

$$P_k = (I - K_k H) P_k^- \qquad (10)$$

Where:
k =0, 1, 2, 3,…, time instants
$\hat{x}_k^-$ =State estimate for an instant k
$\hat{x}_{k-1}^-$ = State in the instant k-1
A = Feedback Matrix
B = Input Matrix
w_{k-1} = Represents the noise inherent to the process
P_k^- =Covariance matrix of the state estimate for the instant k
P_{k-1} = Covariance matrix for the instant k-1.
Q =Covariance matrix of the disturbance in the process
R = Covariance matrix of the disturbance in the observation.
K_k = Filter Gain
z_k = Measure taken from the observer
H = Matrix that relates the state with the measures.
I = Identity matrix

Equation (6) represents the analyzed behavior model system, which contains the state variables to be estimated; determining the relationship between the past state of the system and the current state. Equation (7) represents the estimation of the covariance matrix of the error projecting forward. Equation (8) calculates the Kalman gain, which provides a weighted sum of the prediction and observation, indicating the influence of the error between the estimation and the measurement. Equation (9) represents the updating of the estimation with the measurement, providing a corrected state of the system as the results. Equation (10) shows the correction in the covariance matrix.

$$z_k = H \hat{x}_k^- + v_k \qquad (11)$$

Equation (11) corresponds to the measurements obtained through an external observer of the state variables of the system, *vk* is the value of the noise, which is presented in the measurement of states, and *H*, as previously mentioned, is the matrix that relates the state with the measurements. The matrices *v* and *w* are independent of each other and correspond to a zero-mean white noise presence in the process and in the measurements made, respectively. The matrix *Q* is the covariance associated with the process, which may or may not be diagonal. The matrix *R* is the covariance of the disturbance in the observation, and like matrix *Q*, may or may not be diagonal; if the matrix *R* approaches 0, the gain weighs the discrepancy between the prediction and actual observation with more weight [4]. In practice, the matrices *Q* and *R* can vary in time but in general can be considered as constant [5], [6]. On the other hand, if when changing the value of *R*, it is assigned a small value (in other words, considering that disturbances in the measurements are small), the Kalman gain will be great and greater accuracy will be given to measurements made at calculating the next value of the estimated state. The determination of *Q* is more complicated, since this matrix is responsible for representing the noise in the process and it is generally more difficult to determine because it would be necessary to observe the process directly under analysis.

In the case of the covariance matrix, the initial estimation with which the Kalman filter (see Fig. 4) can be started is the identity matrix, and in the case of the state´s initial value, it can be assigned a value which agrees with the process where the filter is being applied [7].

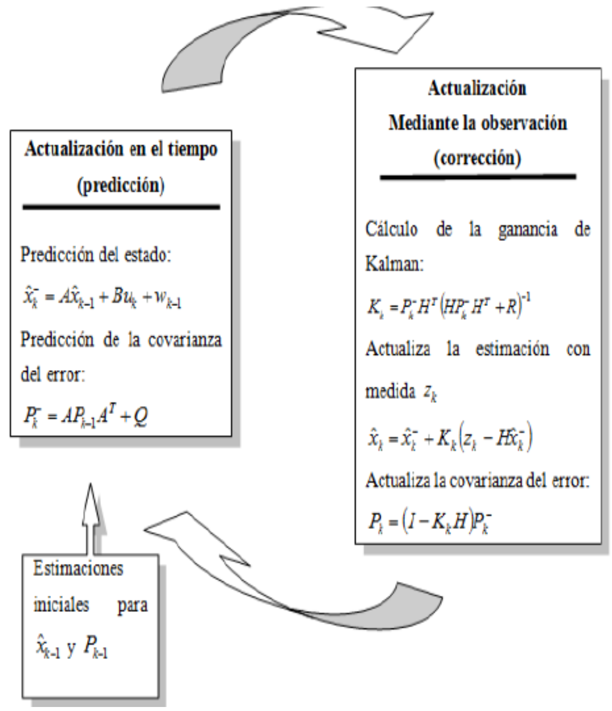


Fig.4. Complete operation of the Kalman Filter, taken from [7]

A. *Implementation of Kalman Filter*

In the case of estimating the position, a model which can be considered as linear is governed by the expression (12)

$$X_k^- = A^* X_{k-1} + B^* X \qquad (12)$$

Where:
$A = 1$
$B = 1$
X_k^- =Estimated position.
X_{k-1} = Previous position.
X =Robotic platform displacement obtained by using odometry.

Equation (12) represents the system defined by odometry, expressed in terms of a single state variable. Considering that it is necessary to obtain the position of each one of the robots, and their location is determined by a matrix of three rows and one column (*x, y, ϴ*) for each of the platforms, then the model will have six state variables as follows:

$$\begin{pmatrix} x_k^-\_r \\ y_k^-\_r \\ \Theta_k^-\_r \\ x_k^-\_a \\ y_k^-\_a \\ \Theta_k^-\_a \end{pmatrix} = \begin{pmatrix} 100000 \\ 010000 \\ 001000 \\ 000100 \\ 000010 \\ 000001 \end{pmatrix} * \begin{pmatrix} x_{k-1}\_r \\ y_{k-1}\_r \\ \Theta_{k-1}\_r \\ x_{k-1}\_a \\ y_{k-1}\_a \\ \Theta_{k-1}\_a \end{pmatrix} + \begin{pmatrix} 100000 \\ 010000 \\ 001000 \\ 000100 \\ 000010 \\ 000001 \end{pmatrix} * \begin{pmatrix} X\_r \\ Y\_r \\ \theta\_r \\ X\_a \\ Y\_a \\ \theta\_a \end{pmatrix}$$

Then:

$$A=\begin{pmatrix}1&0&0&0&0&0\\0&1&0&0&0&0\\0&0&1&0&0&0\\0&0&0&1&0&0\\0&0&0&0&1&0\\0&0&0&0&0&1\end{pmatrix}\quad B=\begin{pmatrix}1&0&0&0&0&0\\0&1&0&0&0&0\\0&0&1&0&0&0\\0&0&0&1&0&0\\0&0&0&0&1&0\\0&0&0&0&0&1\end{pmatrix}$$

$$\hat{X}_k^- = \begin{pmatrix}x_k^- \_r\\ y_k^- \_r\\ \Theta_k^- \_r\\ x_k^- \_a\\ y_k^- \_a\\ \Theta_k^- \_a\end{pmatrix}\quad \hat{X}_{k-1}^- = \begin{pmatrix}x_{k-1}\_r\\ y_{k-1}\_r\\ \Theta_{k-1}\_r\\ x_{k-1}\_a\\ y_{k-1}\_a\\ \Theta_{k-1}\_a\end{pmatrix}$$

Where:

$x_k^- \_r$ =Estimated position X of Robotino2

$y_k^- \_r$ =Estimated position Y of Robotino2

$\Theta_k^- \_r$ =Estimated position Θ of Robotino2

$x_k^- \_a$ =Estimated position X of Robotino1

$y_k^- \_a$ =Estimated position Y of Robotino1

$\Theta_k^- \_a$ =Estimated position Θ of Robotino1

x_{k-1}_r =Previous position X of Robotino2

y_{k-1}_r =Previous position Y of Robotino2

Θ_{k-1}_r = Previous position Θ of Robotino2

x_{k-1}_a = Previous position X of Robotino1

y_{k-1}_a = Previous position Y of Robotino1

Θ_{k-1}_a = Previous position Θ of Robotino1

The state of the system will be initialized in the values obtained from the cameras (Stereoscopic vision system) in the following way:

$$xk\_ = \begin{bmatrix}\mathrm{Xr(3)}\\ \mathrm{Xr(1)}\\ \mathrm{atan(Xr(3)/Xr(1))}\\ \mathrm{Xa(3)}\\ \mathrm{Xa(1)}\\ \mathrm{atan(Xa(3)/Xa(1))}\end{bmatrix}$$

Where:

$\mathrm{Xr(3)}$ =Z-axis position of Robotino2 in space.

$\mathrm{Xr(1)}$ =X-axis position of Robotino2 in space.

$\mathrm{atan(Xr(3)/Xr(1))}$ =Inverse tangent between the axis components according to Robotino2.

$\mathrm{Xa(3)}$ = Z-axis position of Robotino1 in space.

$\mathrm{Xa(1)}$ = X-axis position of Robotino1 in space.

$\mathrm{atan(Xa(3)/Xa(1))}$ = Inverse tangent between the axis components according to Robotino1.

This coordinates' adaptation obtained from the stereoscopic vision system is performed to compare the 3D point data obtained from the odometry. While analyzing the results, the Z-axis of the camera corresponds to the X- axis of the odometry, the Y-axis of the camera corresponds to the X-axis of Robotino, and finally the odometry orientation corresponds to the geometric function inverse tangent between the components already mentioned.

In addition:

$$H=I=\begin{pmatrix}1&0&0&0&0&0\\0&1&0&0&0&0\\0&0&1&0&0&0\\0&0&0&1&0&0\\0&0&0&0&1&0\\0&0&0&0&0&1\end{pmatrix}\qquad X=\begin{pmatrix}X\_r\\ Y\_r\\ \theta\_r\\ X\_a\\ Y\_a\\ \theta\_a\end{pmatrix}$$

Where the elements: X_r, Y_r y θ_r correspond to the displacements performed by Robotino2 in the axes "X", "Y" and "ɵ" and measured by odometry; while X_a, Y_a y θ_a refer to the movement of Robotino1 in the axes "*X*", "*Y*" and "ɵ" measured with the help of odometry.

For values of matrix Q, several tests were analyzed and performed in order to obtain its components experimentally, noticing the values obtained by odometry and making measurements with a tape measure, thus getting the following results:

$$Q=\begin{bmatrix}150^2&0&0&0&0&0\\0&150^2&0&0&0&0\\0&0&50^2&0&0&0\\0&0&0&180^2&0&0\\0&0&0&0&100^2&0\\0&0&0&0&0&50^2\end{bmatrix}$$

In order to obtain the covariance matrix of the disturbance of observation, similarly experimental measurements were performed, obtaining the following results:

$$R=\begin{bmatrix}100^2&0&0&0&0&0\\0&200^2&0&0&0&0\\0&0&0.05235^2&0&0&0\\0&0&0&100^2&0&0\\0&0&0&0&200^2&0\\0&0&0&0&0&0.05235^2\end{bmatrix}$$

Odometry values will be initialized to zero because since it is the first iteration of the algorithm, the robot platform has not done any movement.

At the end of the implemented process, the values of the covariance matrix should be low, because one of the goals of using this filter is to reduce errors and at the same time the covariance of the state vector elements (main diagonal of the matrix P).

IV. TESTS AND RESULTS

Then the system operation tests are performed, as well as results and calculation of location errors in 3D space data through the Kalman Filter. Verification tests of the algorithm's resistance to interference in the presence of factors external to the system, which could affect performance, are realized.

A. Kalman Filter constants test

The purpose of this test was to obtain appropriate results from the Kalman filter constants concerning the covariance matrices of the process and the external observer. These constants, which are difficult to estimate and measure, were heuristically found due to factors in the environment such as light, reflectance, and others (see Fig. 5 and 6).

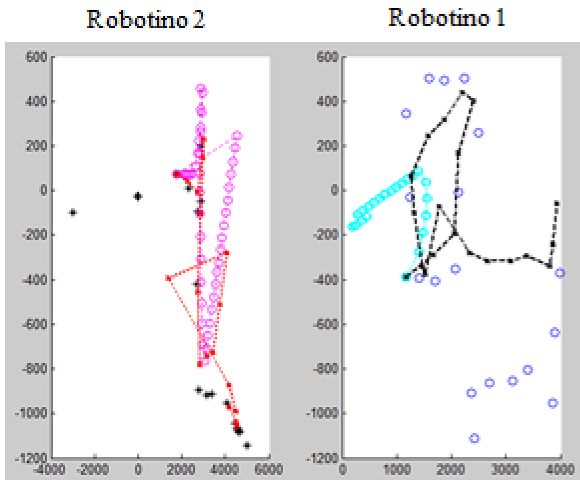


Fig. 5.Graph of localization with erroneous Kalman Filter constants.

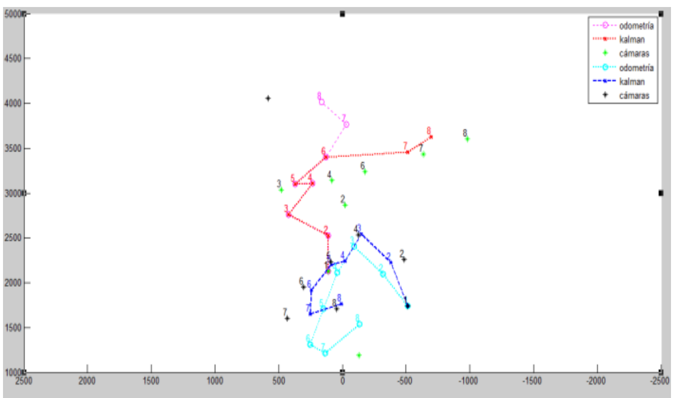


Fig.6. Graphical location with suitable Kalman Filter constants.

B. *Evaluation of localization results.*

Through this test the results obtained by the implemented optimal estimator are evaluated according to reality. In a specific time a group of data is collected point to point, having as real values the measurements taken with a tape measure, and in this way calculating the respective error as shown in Fig. 7 and Tables I and II.

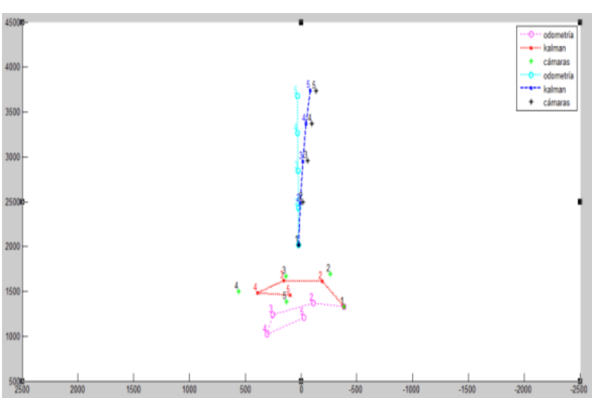


Fig. 7. Test of assessment results point to point (Complete system).

TABLE I
DATA OBTAINED BY ODOMETRY, CAMERAS AND KALMAN FILTER FROM ROBOTINO 2 AND THE CALCULATION OF ERROR

ROBOTINO 2									
CAMERAS [m]		ODOMETRY [m]		KALMAN [m]		REAL [m]		ERROR [%]	
x	y	x	y	X	y	x	y	x	Y
-0,3845	1,3279	-0,3845	1,3279	-0,3845	1,3279	-0,3	1,29	28,17	2,94
-0,2613	1,6904	-0,1115	1,3699	-0,1897	1,6116	-0,18	1,6	5,39	0,72
0,1369	1,6718	0,2518	1,2372	0,1545	1,6236	0,16	1,6	3,44	1,47
0,5599	1,5031	0,3070	1,0241	0,3917	1,4800	0,57	1,35	31,28	9,63
0,1300	1,3900	-0,0283	1,2122	0,0946	1,4595	0,2	1,3	52,7	12,27

TABLE II
DATA OBTAINED BY ODOMETRY, CAMERAS AND KALMAN FILTER FROM ROBOTINO1 AND THE CALCULATION OF ERROR

ROBOTINO 1									
CAMERAS [m]		ODOMETRY [m]		KALMAN [m]		REAL [m]		ERROR [%]	
x	y	x	y	X	y	x	y	x	Y
0,0226	2,0170	0,0226	2,0170	0,0226	2,0170	0,15	1,91	84,93	5,6
-0,0182	2,4935	0,0224	2,4316	0,0074	2,4814	-0,01	2,35	26	5,59
-0,0583	2,9615	0,0272	2,8477	-0,0147	2,9488	-0,14	2,75	89,5	7,23
-0,0958	3,3694	0,0304	3,2641	-0,0442	3,3686	-0,13	3,12	66	7,97
-0,1368	3,7350	0,0287	3,6764	-0,0813	3,7441	-0,18	3,5	54,83	6,67

C. *Test of performance of Kalman Filter due to loss of observer*

This test is done with the objective of showing the resistance to interference of the implemented system at the loss of the observer (stereoscopic vision). The system responds properly in the absence of the observer by monitoring odometry values which still remain. In Table II and Fig. 8 the loss of the observer in the case of Robotino 1 is shown.

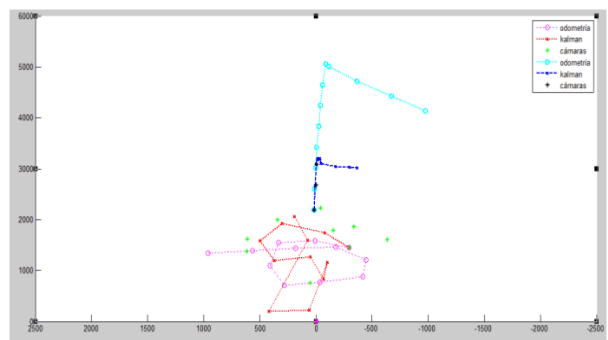


Fig. 8. Test of Kalman Filter Behavior at the loss of observer (Complete System).

TABLE III
DATA OBTAINED BY ODOMETRY, CAMERAS AND KALMAN FILTER FROM ROBOTINO 1 (TEST OF KALMAN FILTER BEHAVIOR AT THE LOSS OF THE OBSERVER).

ROBOTINO1					
CAMERAS [m]		ODOMETRY [m]		KALMAN [m]	
X	y	X	Y	x	Y
0.0142	2.1905	0.0142	2.1905	0.0142	2.1905
-0.0014	2.6808	0.0103	2.6053	0.0060	2.6660
-0.0058	3.0952	0.0019	3.0131	-0.0037	3.0909
--	--	-0.0091	3.4203	-0.0112	3.1751
--	--	-0.0244	3.8325	-0.0184	3.1928
--	--	-0.0409	4.2421	-0.0236	3.1958
--	--	-0.0598	4.6508	-0.0282	3.1962
--	--	-0.0871	5.0625	-0.0361	3.1969
--	--	-0.1177	5.0120	-0.0429	3.1053
--	--	-0.3650	4.7208	-0.1791	3.0395
--	--	-0.6747	4.4233	-0.3003	3.0251
--	--	-0.9746	4.1333	-0.3681	3.0238

V. CONCLUSIONS

The existence of uncertainty in the odometry makes the determination of the process's covariance matrix difficult to find, so it uses the heuristic method.

Each pixel in the image provides some information about an elemental region of it. The information in the color images corresponds to the intensity of each one of the RGB format components, which allows the association of each pixel with a distinctive label for segmenting the object of interest.

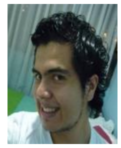
The values provided by the stereoscopic system present some fluctuation due to the brightness in the work environment. Calibration should be done, preferably in the same environment, and realized meticulously, because the results are the basis to obtain the 3D point in space. Likewise, it influences the calculation of the observer's covariance matrix, which was experimentally determined.

The Kalman Filter allows estimation of the optimal position of the robot platforms, with the cameras as an external observer and odometry as the process. The latter can be considered as a stochastic system observed through a sensor (camera) that presents a certain level of noise, which is normally in the images.

The implemented location system is resistant to interference because it supports disturbances produced by a loss of the observer, either by being out of sight of the cameras, by momentary occlusions, or by distances which exceed the system design to obtain 3D points. This is because odometry is able to support the location system by itself during a short time interval.

REFERENCES

[1] R. Barca, E. López, L.M. Bergasa, S. Álvarez, M. Ocaña, I. Novo,"LocalizaciónMulti-Robot Colaborativa basada en el Método deMonte-Carlo en Robots Asistenciales", Dept. de Electrónica, Universidad de Alcalá, Madrid, España, 2006.
[2] A. Fuentes, "Localización y Modelado Simultáneos en ROS para la plataforma Robótica Manfred", Dept. de Ingeniería de Sistemas y Automática, Universidad Carlos III de Madrid, Madrid, 2011.
[3] P. Hilman,"Camera Calibration and Stereo Vision", Edimburgo, 2005.
[4] P. Rodríguez, "Aplicación del Filtro de Kalman al Seguimiento de objetos en secuencias de imágenes", Universidad Rey Juan Carlos, España, 2003.
[5] G. Welch , G. Bishop, "An Introduction to the Kalman Filter", Dept. of computer Science, University of North Carolina, Chapel Hill, EEUU., July, 2006.
[6] A. Solera, "El Filtro de Kalman", Banco Central de Costa Rica, División Económica, Dept. de Investigaciones Económicas., Julio, 2003.
[7] D. Pozo, N. Sotomayor, J. Rosero, L. Morales, "Medición de Ángulos de Inclinación por Medio de Fusión Sensorial aplicando Filtro de Kalman", EPN, Quito, 2013.
[8] Guang-yao Jin, Xiao-Yi Lu, Myong-Soon Park, "An indoor localization mechanism using active RFID tag", IEEE International Conference onSensor Networks, Ubiquitous, and Trustworthy Computing, 2006.
[9] Krishna Chintalapudi, AnandPadmanabha, Venkata N. Padmanabhan, "Indoor localization without the pain", Proceedings of the sixteenth annual international conference on Mobile computing and networking, 2010.
[10] Yingying Chen, Francisco J., Trappe W., Martin R.P., "A Practical Approach to Landmark Deployment for Indoor Localization", 3rd Annual IEEE Communications Society on Sensor and Ad Hoc Communications and Networks. 2006.
[11] Chenavier, F., Crowley, J.L., "Position estimation for a mobile robot using vision and odometry", IEEE International Conference on Robotics and Automation, 1992.
[12] Larsen, T.D., Hansen, K.L., Andersen, N.A., Ravn, O., "Design of Kalman filters for mobile robots; evaluation of the kinematic and odometric approach", IEEE International Conference on Control Applications, 1999.
[13] R.-C. Weber, M. Bellenberg, Robotino Manual, Festo Didactic GmbH & Co. KG, Denkendorf, Germany, 2010.



Cristian Revelo, was born in Ibarra-Ecuador the 26th of March, 1990. He studied at Diocesan Bilingual High School, where he was chosen as the best Graduated of the academic year 2006-2007, getting the High School Degree in Physics and Mathematics. He studied at "Escuela Politécnica Nacional" the career of Electronics and Control. (cris_andres115@hotmail.com).

María Fernanda Trujillo, was born in Riobamba-Ecuador the 15th of de February, 1990. She studied at "Santa Mariana de Jesús" High School, where she was chosen as Standart Bearer of the High School Flag in the academic year 2006-2007, and she got the High School Degree in Physics and Mathematics. In 2014, she finished the career of Electronics and Control Engineering at "Escuela Politécnica Nacional". (maryfertg@gmail.com).

David Pozo, was born on april 4, 1985 in Quito, Ecuador. His high school studies were made in "Juan Pío Montúfar" high school, and his college studies in the National Polytechnic School in Quito, Ecuador obtaining a degree in bachelor of Electronic and Control in the year 2010. In the year 2012 he obtained a Master's degree in Automatics and Robotics in the Cataluña Polytechnic University, Barcelona, Spain. He worked as a full time teacher in the National Polytechnic School in the Automation and Control area. Nowadays he works as a teacher in University of The Americas (UDLA) (david_mh4485@yahoo.com) .

Andres Rosales was born in Quito, Ecuador. He completed his studies at the Escuela Politécnica Nacional (EPN), where he graduated in Electronics and Control Engineering (2001). He completed his Ph.D. in Control Systems Engineering (2009) at the Institute of Automation (INAUT) of the National University of San Juan, Argentina; also conducted a research stay at the Institute for Real-Time Systems (RTS) of the University of Hannover (2007-2008), Germany as a DAAD scholarship (German Academic Exchange Institute). He is currently the General Technical Coordinator in the Council of Evaluation, Accreditation and Quality Assurance in Higher Education - CEAACES, and Editor of the EPN Journal. He has worked in R&D Departments of CENACE, CELEC EP and the Ministry of Electricity and Renewable Energy. From 2012 to 2014 he was General Coordinator of the Ecuadorian Network of Universities and Polytechnics for Research and Post Graduate Studies - REDU. Also, he holds the position of Principal Professor in the Department of Automation and Control at EPN. (andres.rosales@epn.edu.ec).