Robot pose estimation by means of a stereo vision system

J. SOGORB, O. REINOSO, A. GIL, L. PAYA
Automation, Robotics and Computer Vision
Systems Engineering and Automation Department
University Miguel Hernández
Av. Universidad 0, 03202 - Elche (Alicante)
SPAIN
{jsogorb, o.reinoso, arturo.gil, lpaya}@umh.es http://isa.umh.es/arvc/

Abstract: Mobile robots are characterised by their capacity to move autonomously in an environment that is either known or unknown or only partially known. Their uses and applications are wide and are incorporated into a great many fields including underground and submarine work, space missions, security systems, military applications, and many more. It is for this reason that that a mobile robot is rarely fitted with only one sensor to carry out all of its multiple tasks, being much more frequent the use of various sensors combined within the system that complement one another to complete their different functions. In this way it is possible to find robots where estimation of position1 and the updating of the map is carried out by video cameras or laser scanners, while obstacle detection is achieved using sonar. In this respect it is important to highlight the close relationship that exists between the problem of position estimation and that of the construction of a map of the surroundings, with exact localisation of the robot necessary to be able to carry out map construction and vice versa. In this work we focus solely on the problem of localisation, comparing different estimation algorithms of the trajectory taken by a robot from the observations and readings obtained by the robot itself. In our problem, we will work with images taken by a stereoscopic vision system of uncalibrated cameras, we will assume that the movement of the robot is on a flat surface and we will use natural landmarks. As we will see, the information obtained from this type of sensor allows a robust estimation of movement taken between each pair of observations without the need to use the information from the robot's propioceptive sensors. The solution of this problem, known as visual odometry, is critical within the majority of subsequent navigation processes.

Key-Words: Visual odometry, mobile robot, stereo vision, position estimation, natural features.

1. Introduction

Mobile robots are characterised by their capacity to move autonomously in an environment that is either known or unknown or only partially known. Their uses and applications are wide and are incorporated into a great many fields including underground and submarine work, space missions, security systems, military applications, and many more. It is for this reason that a mobile robot is rarely fitted with only one sensor to carry out all of its multiple tasks, being much more frequent the use of various sensors combined within the system that complement one another to complete their different functions. [1].

In this way it is possible to find robots where estimation of position[1] and the updating of the map is carried out by video cameras or laser scanners, while obstacle detection is achieved using sonar [2] [9]. In this respect it is important to highlight the close relationship that exists between the problem of position estimation and that of the construction of a map of the surroundings, with exact localisation of the robot necessary to be able to carry out map construction and vice versa.

In general terms, determining the position of a mobile robot is equivalent to finding the components of movement (tx, ty, tz) and rotation (θx, θy, θz) of the system of coordinates supportive of the robot (and therefore mobile) with respect to an absolute system. Specifically, in this work a bi-dimensional case is considered (by far the most common application of mobile robots today), where the robot moves with three possible degrees of freedom. In this way, the problem is

1. Throughout this work the expression "estimation of position" is used to refer to both the obtainment of the position and to the orientation of the vehicle.

reduced to finding the three values (tx, ty, θ) associated to the mobile system of the vehicle, where (tx, ty) represent its position and θ represents its orientation.

The majority of mobile robots are fitted with encoders on the movement axles that allow constant localisation estimation through the use of a locomotion model. However, this estimation is not sufficiently exact for the majority of applications. The reason is not due to the errors that can be made, but is more a result of the accumulation of these errors throughout the navigation process, something that means that the region of uncertainty associated with the robot's position and orientation increases progressively as the robot moves [3]. Because of this, each time certain limits are passed, the robot needs the help of an "external" positioning system to reduce this uncertainty [4].

In this work a technique to estimate the actions carried out by the robot from the stereo observations obtained throughout the trajectory is presented. To achieve this, only the geometric information from the obtained observations will be used. As mentioned previously, the only supposition made is that the robot always moves on a flat surface, so we have 3 degrees of freedom.

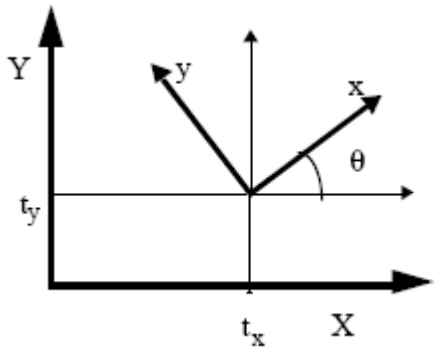

Fig. 1: Inner reference coordinate system of robot

In order to quantify the goodness of these estimations different criteria are used that can be interpreted through some statistical indexes. In this way, the estimation problem can be formulated as a problem of optimization of a determined index. Amongst the most used criteria are least squares and maximum verisimilitude [5], [6], [7], [10].

In the criteria of least squares we try to obtain an estimation of the vector of L parameters that minimize the index:

$$J = \|y - \hat{y}\| = (y - \hat{y})^T (y - \hat{y}) = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 \quad (1)$$

being $\mathbf{y} = (y_1, y_2, \ldots, y_n)^T$ a collection of N measurements and $\hat{\mathbf{y}} = (\hat{y}_1, \hat{y}_2, \ldots, \hat{y}_n)^T$ values calculated from the model adapted to the system. For example, for a lineal static system we have:

$$\hat{y} = M \cdot \hat{\theta} \quad (2)$$

where **M** is a matrix of N rows and L columns, with $N \geq L$.

The estimation of least squares also allows the grading scale of errors or residues, in this case using the index:

$$J = (y - \hat{y})^T W (y - \hat{y})$$
$$J = \sum_{i=1}^{n} w_i (y_i - \hat{y}_i)^2 = \sum_{i=1}^{n} w_i e_i^2 \quad (3)$$

where e_i is the error or residue corresponding to the measurement yi, and **W** is a diagonal matrix of consideration whose elements are w_i, i=1, …, N.

The criteria of maximum verisimilitude is based on the definition of a function L(y,θ) denominated as "verisimilitude" that is normally the function of conditional probability p(y|θ). Supposing we have a group of independent measurements $y = (y_1, y_2, \ldots, y_n)^T$, we try to find the parameters that make the measurements have a greater probability of occurring.

Bearing in mind that

$$L(y,\theta) = p(y \mid \theta) = \frac{p(y,\theta)}{p(\theta)} \quad (4)$$

The problem can be resolved from knowledge of the function of probability density grouping p(y, θ) and some previous knowledge of θ that permits the establishment of the function of probability density p(θ).

It is important to note that when the noise associated with the measurements is modelled as Gaussian white noise with invalid media and diagonal covariance matrix, it can be demonstrated that the estimator of maximum verisimilitude is

equivalent to the estimator of non-linear least squares given by the equation (3) [7], [8], [11].

As well as these, there also exists other estimation criteria based on a posteriori conditional probability p(θ | y) [5], [7].

2. Estimation of movement

The objective consists in determining, in each instant, the transformation matrix related to ***A*** that indicates the position and orientation of the stereo pair in an instant $t + \Delta t$ with respect to the stereo pair in an immediately previous instant t.

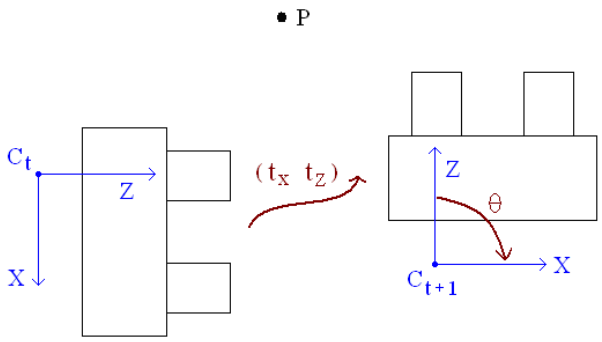

Fig. 2: Movement & rotation of the camera on a plane

The matrix of transformation ***A*** presents the following structure:

$$\begin{bmatrix} {}^{Ct}X_P \\ {}^{Ct}Z_P \\ 1 \end{bmatrix} = A \cdot \begin{bmatrix} {}^{Ct+1}X_P \\ {}^{Ct+1}Z_P \\ 1 \end{bmatrix}$$

$$\begin{bmatrix} {}^{Ct}X_P \\ {}^{Ct}Z_P \\ 1 \end{bmatrix} = \begin{bmatrix} \cos\theta & sen\theta & t_X \\ -sen\theta & \cos\theta & t_Z \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} {}^{Ct+1}X_P \\ {}^{Ct+1}Z_P \\ 1 \end{bmatrix} \quad (5)$$

Given that we have 4 unknowns and each point **P** contributes 2 equations, we need to know, in each pair of consecutive instants, the X and Z coordinates of two points, in a way in which the system of equations has a single solution.

In this way, the system of equations to resolve is the following:

$$\begin{bmatrix} {}^{Ct+1}X_{P1} & {}^{Ct+1}Z_{P1} & 1 & 0 \\ {}^{Ct+1}Z_{P1} & -{}^{Ct+1}X_{P1} & 0 & 1 \\ {}^{Ct+1}X_{P2} & {}^{Ct+1}Z_{P2} & 1 & 0 \\ {}^{Ct+1}Z_{P2} & -{}^{Ct+1}X_{P2} & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} a \\ b \\ t_X \\ t_Z \end{bmatrix} = \begin{bmatrix} {}^{Ct}X_{P1} \\ {}^{Ct}Z_{P1} \\ {}^{Ct}X_{P2} \\ {}^{Ct}Z_{P2} \end{bmatrix} \quad (6)$$

denoting a = *cos*θ and b = *sen*θ .

To be able to obtain the unknowns (parameters of the matrix ***A***) it is necessary to recognise two points in two pairs of stereo images taken in two consecutive instants.

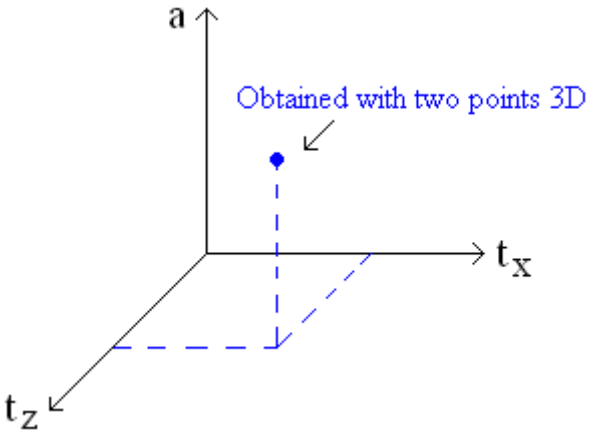

Fig. 3: Space of coordinates (a , t_X , t_Z)

It is important to note that each pair of points allows us to determine the parameters of movement a = *cos*θ, t_X y t_Z, which represent a point in the space of coordinates (a , t_X , t_Z).

In a situation in which we have three points, then we could resolve (*3 2*) = 3 systems of four equations, with the result of 3 points in the space (a, t_X, t_Z).

Generally speaking, if we know the coordinates XYZ (in reality only the X and Z coordinates are necessary, given that we have done a previous transformation of coordinates and, after that, we assume the movement of the robot is flat) of ***N*** points in two consecutive instants, we can resolve (*N 2*) systems of 4 equations, with the result of a cloud formed by (*N 2*) points in the space of coordinates (a , t_X , t_Z).

Each point represents a single solution to the problem of movement determination of the stereo pair, and by extension the robot.

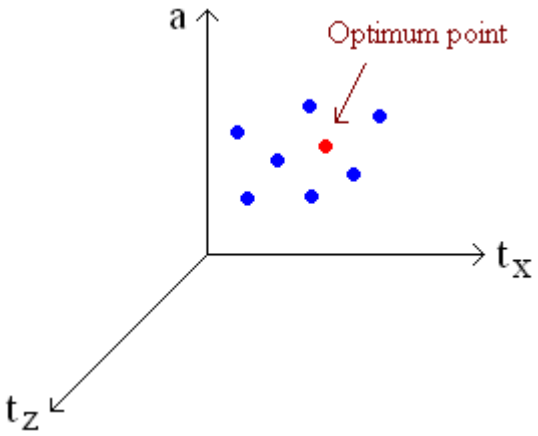

Fig. 4: Optimum point (a , t_X , t_Z)

Once this cloud of points is obtained, an algorithm must be used that allows us to adjust this cloud of points to a model, that is to say, that provides us with the optimum point (a , t_X , t_Z). This point will represent the incremental movement of the robot between two consecutive instants.

2.1. Selection of the optimum point (t_x, t_y, θ)

In this section we will implement and compare three methods to select the optimum point from the cloud of points obtained as a result of resolution of the system of equations studied in the previous section.

RANSAC Algorithm

In this section we will describe a general robust algorithm known as ***RAN****dom* ***SA****mple* ***C****onsensus* (RANSAC), specific to the case in question in which we wish to obtain three parameters (a , t_X , t_Z).

To give more detail, the steps of the RANSAC algorithm are the following:

1) Select a random point Pi = (a, t_X , t_Z)$_i$.

2) Determine the group of points S that are located within the sphere drawn by radius T centred on P_i.

$$\left\| P_i \;\; P_j \right\| < T \qquad \forall j \neq i \qquad (7)$$

3) If the number of points in S, that we shall call s, is above a threshold t, then this point will be stored.

4) If the number of points in S is less than t, then this point will be rejected.

5) The four previous steps are repeated until the point that meets the conditions of step 3) is found.

Figure 4 shows graphically the steps of RANSAC algorithm.

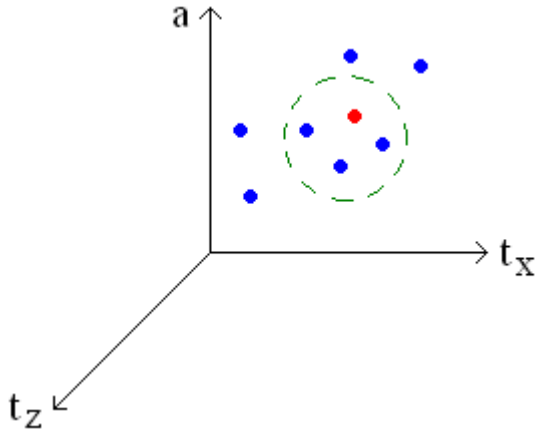

Fig. 5: RANSAC Algorithm

Selection of the threshold

The threshold is selected in a way that with probability α the point will be an "inlier". This calculation needs a probability distribution for the distance of an "inlier" from the model. In practice, this distance is usually chosen empirically. However, if it is assumed that the measurement error is Gaussian with media zero and typical deviation σ, then

$$t^2 = F^{-1}{}_m(\alpha) \cdot \sigma^2 \qquad (8)$$

being F a chi-squared distribution with m degrees of freedom. The values calculated for the case of α = 0.95 are:

- In the adjustment of a line or a fundamental matrix, we have m = 1, then $t^2 = 3.84\sigma^2$.
- In the adjustment of a homography or the matrix of a camera, we have m = 2, then $t^2 = 5.99\sigma^2$.

Number of readings

It is not necessary to treat all of possible readings. However, a sufficiently high number of readings must be taken to guarantee, with probability ***p***, that at least one of the random groupings of ***S*** points is free of variant points. If we suppose that ***ε*** is the probability that a point is variant, then this value will be a percentage of the total number of points, it can be proven that for the given probabilities ***p*** and ***ε***, the size of the number of readings to be taken is given by the expression,

$$N = \log(1-p) / \log\left(1-(1-\varepsilon)^{S}\right) \qquad (9)$$

Normally the value of ε will not be known beforehand; therefore it is necessary to use an iterative algorithm that readapts the values of ***ε*** and ***N*** at the same time as new readings are obtained.

Mean

A quick and easy way to obtain the optimum point consists in calculating the mean of the points (a, t_X , t_Z) recorded.

For the parameters that determine the vector of movement, we simple work out:

$$t_X = \frac{1}{N}\sum_{i=1}^{N} t_{X_i} \qquad t_Z = \frac{1}{N}\sum_{i=1}^{N} t_{Z_i} \qquad (10)$$

However, the parameter "a" of each point only varies between -1 and 1, so we don't calculate the mean of the different values of "a", but we first discover the value of angle θ and then determine the mean of the angles.

$$\theta_i = \arccos(a_i) \quad \rightarrow \quad \theta = \frac{1}{N}\sum_{i=1}^{N} \theta_i \qquad (11)$$

Median

Just as with the mean, the median quickly and easily provides the optimum value of the parameters that represent the robot's movement.

As we know, the median of a distribution (group of values) is the value that equally divides the distribution, 50% above and the other 50% below.

Therefore, to discover the median we simple have to place the different values from each parameter in a vector, order them and select the central value.

$$\left(tx_1 \quad tx_2 \quad \ldots \quad tx^* \quad \ldots \quad tx_{N-1} \quad tx_N\right) \qquad (12)$$

$$\left(tz_1 \quad tz_2 \quad \ldots \quad tz^* \quad \ldots \quad tz_{N-1} \quad tz_N\right) \qquad (13)$$

$$\left(a_1 \quad a_2 \quad \ldots \quad a^* \quad \ldots \quad a_{N-1} \quad a_N\right) \qquad (14)$$

2.2. Movement updating

The optimum parameters (a , t_X , t_Z) obtained previously represent the incremental movement of the robot between two instants or consecutive captures.

Evidently, this incremental movement is expressed with respect to the system of coordinates of the camera in the immediately previous instant. Given that the system of coordinates of the camera moves with the robot, we must express the incremental movement calculated in each iteration with respect to a system of fixed reference.

In agreement with equation (5),

$$\begin{cases} X_P{}^{t} = t_X + X_P{}^{t+1} \cdot \cos\theta + Z_P{}^{t+1} \cdot sen\theta \\ Z_P{}^{t} = t_Z + Z_P{}^{t+1} \cdot \cos\theta - X_P{}^{t+1} \cdot sen\theta \end{cases} \qquad (15)$$

Then, the equations that let us obtain the absolute parameters with respect to a system of fixed reference, are the following:

1) $$t_X{}^{t+1} = t_X{}^{t} + \Delta t_X \cdot \cos\theta^t + \Delta t_Z \cdot sen\theta^t \qquad (16)$$

2) $$t_Z{}^{t+1} = t_Z{}^{t} + \Delta t_Z \cdot \cos\theta^t - \Delta t_X \cdot sen\theta^t \qquad (17)$$

3) $$\theta^{t+1} = \theta^t + \Delta\theta \qquad (18)$$

where,

- (a , t_X , t_Z)$^{t+1}$ represent the position and orientation of the robot in the present instant with respect to system of fixed reference.

This system of fixed reference is the system of coordinates of the camera in the initial instant, that is, the moment in which the robot begins to move.

- $(a , t_X , t_Z)^t$ represent the position and orientation of the robot instant immediately previous with respect to a system of fixed reference.

- $(\Delta a , \Delta t_X , \Delta t_Z)$ represent the position and orientation of the robot in the present instant with respect to the system of previous coordinates. That is, they represent the incremental movement, obtained for each iteration by the procedure described above.

3. Experiments

In this section we are going to put forward the results obtained for each of the three methods, visualizing the optimum point chosen from the cloud of points generated in each iteration, as well as a comparison between the estimated trajectory of the robot and the real trajectory.

3.1. Algorithm RANSAC

The parameters that influence the correct functioning of the RANSAC algorithm are:

- Parameter ***T***: this parameter determines the radius of the sphere that contains a sufficiently large number of points.

- Parameter ***t***: this parameter determines the minimum number of points that the sphere, defined by the above parameters, must contain to be able to consider that the point that occupies the centre is the optimum.

After numerous experiments it was concluded that the most appropriate value for the previous values is:

T = 0.01 t = 3 points

With these values results were obtained such as those shown in figure 6.

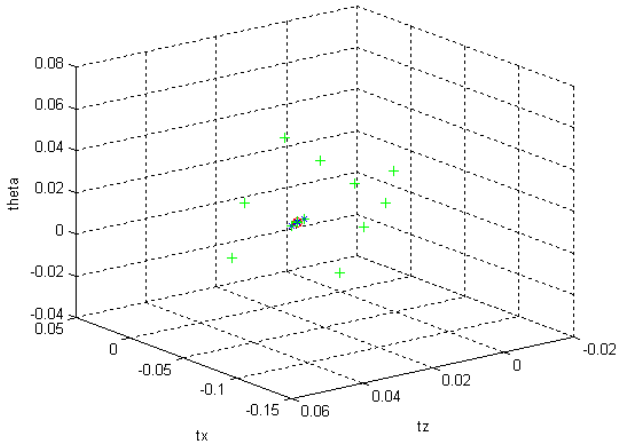

Fig. 6: Points in the space of parameters (tx, tz, theta)

The key used in the graphs is:

+ Points obtained after the resolution of the proposed system of equations.

x Points found within the sphere drawn by radius T.

O Optimum point. Represents the centre of the sphere described above.

To give a more complete vision of the estimation of movement process, the following charts show the trajectory followed by the robot during a sequence of images. Various experiments were carried out, trying different trajectories: a straight line and a curved line.

In all of the charts the estimated trajectory is represented in blue, while the real trajectory appears in red.

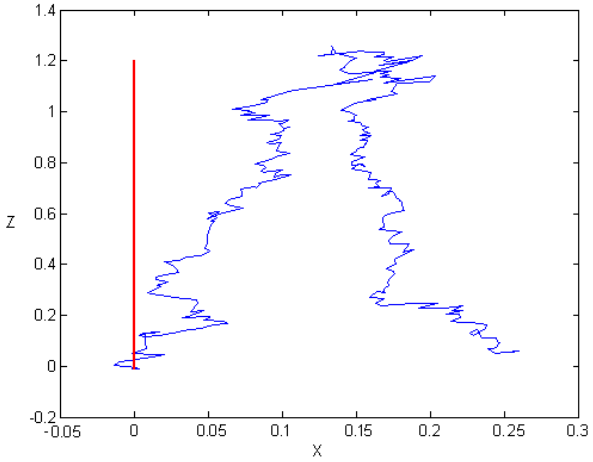

Fig. 7: Estimation of straight line trajectory (RANSAC)

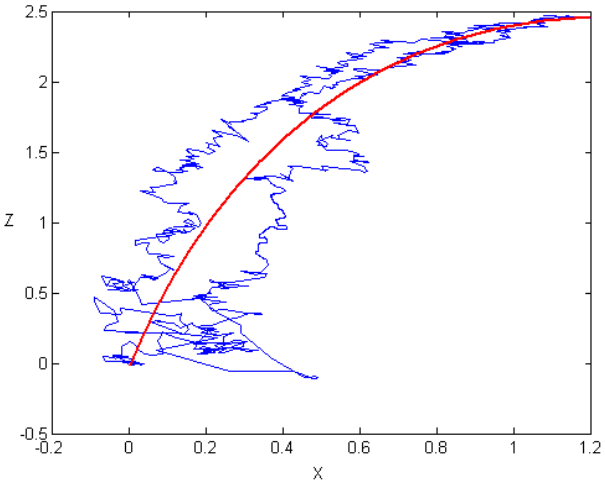

Fig. 8: Estimation of curved trajectory (RANSAC)

3.2. Mean

As we know, the mean method consists in calculating the arithmetical mean of each of the parameters corresponding to the cloud of points generated in the resolution of the proposed system of equations.

In this case, the mean of all the points is calculated. It seems evident that this method will function correctly when the points are grouped closely together and that it will produce incorrect results when there are deviant points separated from the principle grouping, given that these points are also taken into consideration when calculating the mean. Therefore, it is easy to arrive to the conclusion that the mean method produces worse results than the RANSAC method. However, the principal advantage of this method is its speed.

Below are shown the results obtained in the estimation of robot movement in the same trajectories as before, using the mean method.

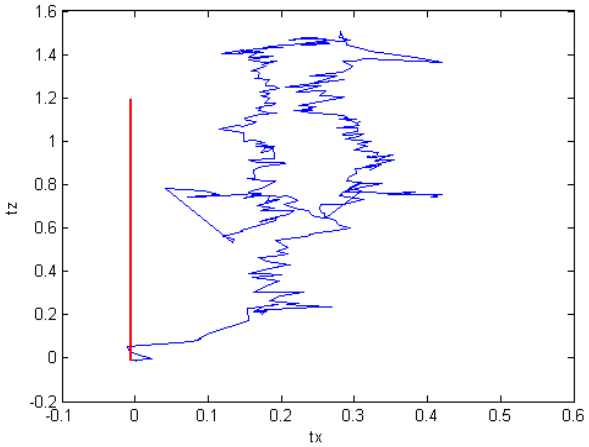

Fig. 9: Estimation of straight line trajectory (mean)

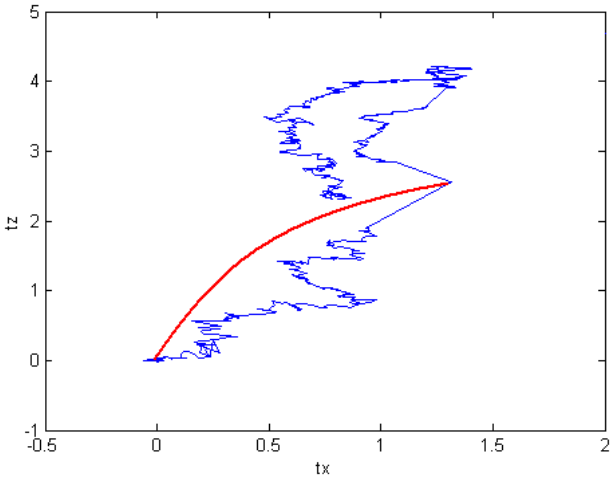

Fig. 10: Estimation of curved trajectory (mean)

3.3. Median

In this case, as with the mean, all of the points are taken into consideration. Therefore, just as before, this method will function correctly when the collection of points is grouped closely together and will produce incorrect results when variant points exist situated far from the principal grouping, because these points are also considered when the median is calculated.

Therefore, we can conclude that the median method produces worse results than the RANSAC method, just the same as the mean. However, the principal advantage of this method is also its speed.

The results obtained were as follows:

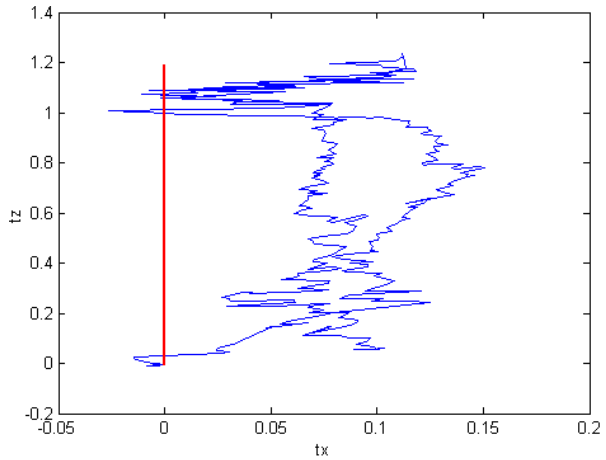

Fig. 11: Estimation of straight line trajectory (median)

This figure shows that median method generates less error in horizontal axis that mean method. This result is similar for vertical axis, as next figure shows.

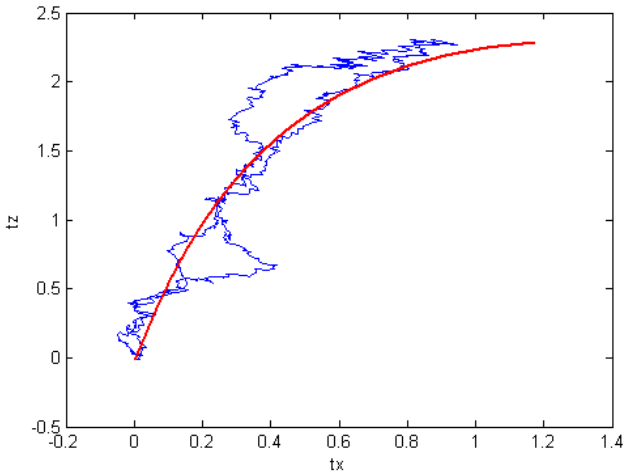

Fig. 12: Estimation of curved trajectory (median)

3.4. Comparison of methods

In this section we will compare the three above methods, analyzing the degree of precision obtained and the computational cost for each.

- To evaluate the precision of each method, we will calculate the root mean square error in the two trajectories, straight and curved.

- To measure the computational cost of each method we will estimate the execution time of an iteration of the algorithm, with the result expressed in milliseconds.

In this way we will have quantitative criteria to help us select one of the three proposed methods. Firstly, we calculate the root mean square error, for which we will need to know the error, that is to say, the difference between the estimated trajectory and the real trajectory. After several experiments, the following results were obtained:

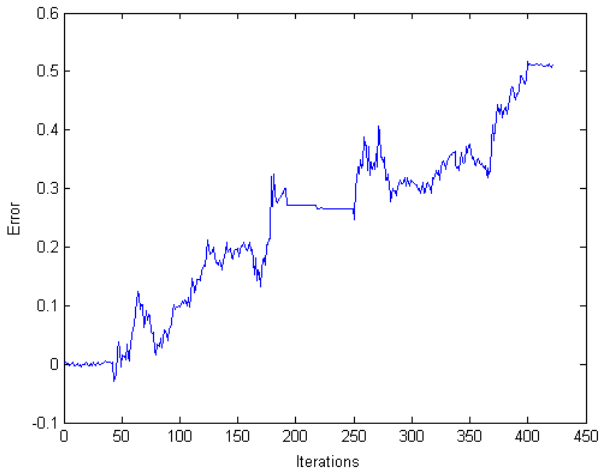

Fig. 13: Error estimation of line trajectory (RANSAC)

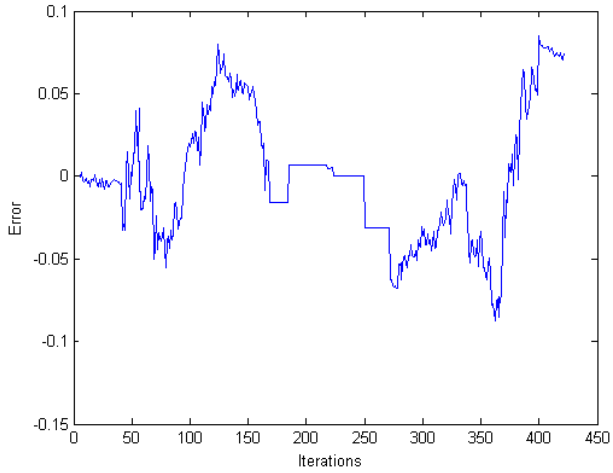

Fig. 14: Error estimation of curved trajectory (RANSAC)

For the mean method, the results obtained were the following:

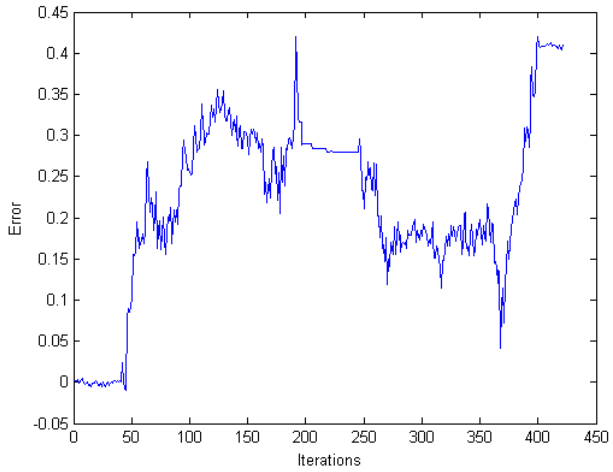

Fig. 15: Error estimation of line trajectory (mean)

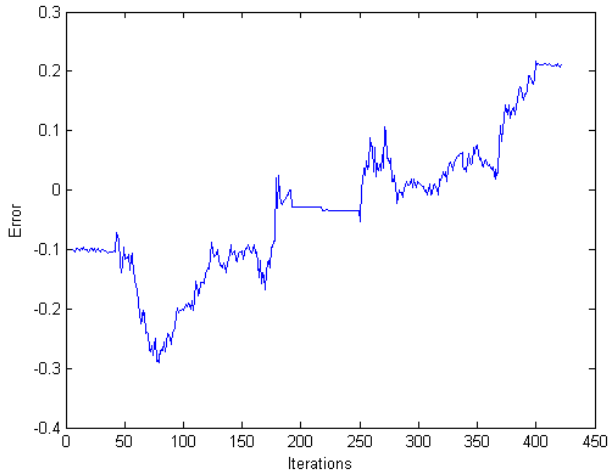

Fig. 16: Error estimation of curved trajectory (mean)

For the median method, the results obtained were the following:

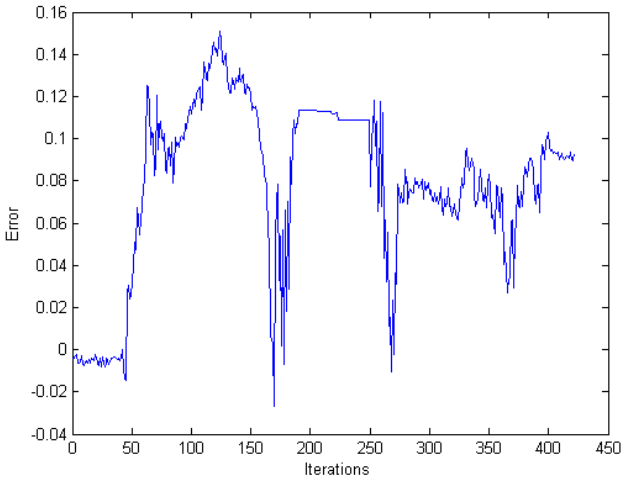

Fig. 17: Error estimation of line trajectory (median)

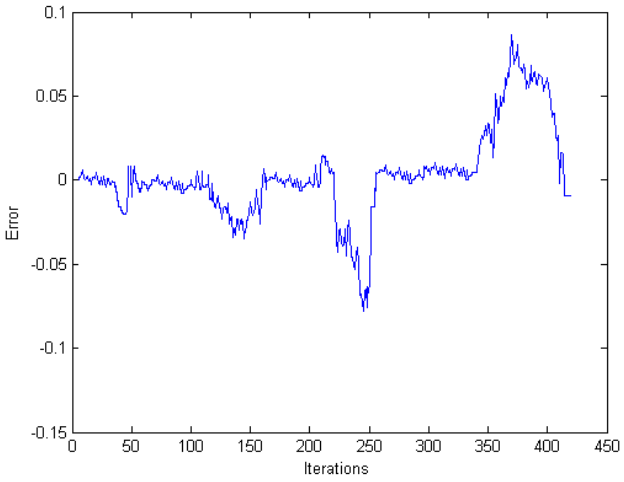

Fig. 18: Error estimation of curved trajectory (median)

Once the error committed in each iteration or reading instant is obtained, we calculate the root mean square error.

$$Error = \frac{1}{N}\sum_{i=1}^{N}\left(x_{estimated_i} - x_{real_i}\right)^2 = \frac{1}{N}\sum_{i=1}^{N} e_i^{\,2} \qquad (19)$$

The results were:

Method	Straight line	Curve
Ransac	0.0235	0.0453
Mean	0.0782	0.155
Median	0.0078	0.0248

Table 1: Root mean square error

The obtained results are also represented in a bar chart in Figure 19.

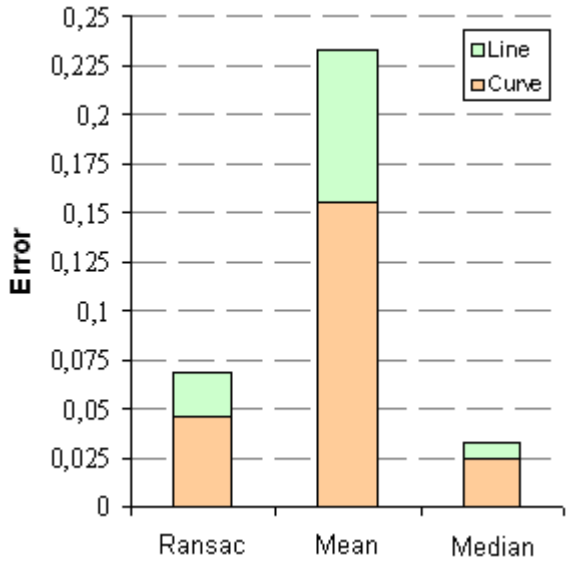

Fig. 19: Comparison of root mean square error

As can be seen, the Ransac and median methods present a root mean square error less than the mean method, with the median method producing the lowest error.

Finally, the execution time per iteration for each of the methods is as follows:

Method	Time (ms)
Ransac	26.366
Mean	19.715
Median	19.002

Table 2: Execution time per iteration

As we can see the lowest computational costs were obtained by the median method followed by the mean.

Therefore, we can conclude that the median is the best method of the three due to its more exact precision and its lower computational cost.

4. Conclusions and future works

From the obtained results the conclusion was arrived at that it is possible to successfully achieve estimation of movement in three different ways: Ransac, mean and median.

Comparing the three proposed algorithms with the previously mentioned analysis criteria, the median method provided the best results, followed by Ransac and the mean method.

Due to the satisfactory results obtained, it is feasible to use this system to complement the internal odometric sensors of the robot in those situations in which the said system does not turn out to be sufficiently precise.

The successes achieved in the development of the system throughout the Project permit its incorporation to the more complex algorithms of SFM y SLAM.

Recuperation of the structure from movement is a typical problem in computerised vision, one that has traditionally been bypassed with the extensive use of multiple vista geometry and of numerical techniques of robust estimation.

In mobile robots it is a double problem, localisation and simultaneous map construction. In both cases two linked estimation problems exist:

- In SFM (*structure from motion*), they are the recuperation of the scene structure and the movement of the camera.
- In SLAM (*simultaneous localization and mapping*) they are the map construction and the self localisation of the mobile robot within this map.

Therefore, an interesting work for the future would be the integration of the estimation of movement algorithm that we have developed, to SFM y SLAM processes, constituyendo una de las múltiples parts que los componen.

Acknowledgements:

This work has been supported by the Spanish Government (Ministerio de Ciencia y Tecnología). Project: "*Sistemas de percepción visual móvil y cooperativo como soporte para la realización de tareas con redes de robots*". Ref: DPI2007-61197

References:

[1] A. Ollero, A. Simon, F. Garcia y V. Torres. "Integrated Mechanical Design of a New Mobile Robot". *Proc. SICICA´92 IFAC Symposium*. Ed. A. Ollero y E.F. Camacho. Pergamon Press, 1993.

[2] V. Torres, J.A. Garcia-Fortes, A. Ollero, J. Gonzalez, A. Reina. "Descripción del Sistema Sensorial del VAM-1". *3º Congreso Nacional de la Asociación Española de Robótica*. Zaragoza, 1993.

[3] C.M. Wang. "Location Estimation and Uncertainty Analysis for Mobile Robots", *IEEE Int. Conf. On Robotics and Automation*, pp. 1230-1235, 1988.

[4] R.C. Smith, P. Cheeseman. "On the Representation and Estimation of Spatial Uncertainty". *The International Journal of Robotics Research*. Vol 5, No 4, 1986.

[5] A. Gelb. "Applied Optimal Estimation". MIT Press, 1974.

[6] L. Matthies, S. Shafer (1987). "Error Modeling in Stereo Navigation". *IEEE Int. Journal of Robotics Research*, vol.3, no.3, 1987.

[7] A. Ollero. "Control por Computador. Descripción Interna y Diseño Optimo". Ed. Marcombo, 1991

[8] J.D. Tardos. "Integracion Multisensorial para Reconocimiento y Localización de Objetos en Robotica". *Tesis Doctoral. Universidad de Zaragoza*, 1990.

[9] Armida González Lorence, Mayra P. Garduño Gaffare and J. Armando Segovia De Los Ríos, "Geometry-Projective Visual LandMarks for Robot Localization", *WSEAS Transactions on Computers*, 2(5), pp. 463-468, February 2006. ISSN 1109-2750.

[10] "Adaptive visual servoing and force control fusion to track surfaces". J. Pomares, G. J. García, L. Payá. F. Torres. *WSEAS Transactions on Systems*. Vol. 5. Num. 1. pp. 25-32. 2006. ISSN: 1109-2777.

[11] C. Massacci, A. Usai, P. Di Giamberardino, "A Radio Connected Intelligent Motor Control Board for Mobile Robotic Applications", *WSEAS Transactions on Circuits and Systems*, Iss. 8, Vol. 5, pp.1259-1265, 2006.