

# Feature Selection Criteria for Real Time EKF-SLAM Algorithm

**Fernando Auat Cheein, Gustavo Scaglia, Fernando di Sciasio and Ricardo Carelli**

Instituto de Automatica, National University of San Juan, San Juan, Argentina.  
fauat@inaut.unsj.edu.ar

**Abstract:** This paper presents a selection procedure for environment features for the correction stage of a SLAM (Simultaneous Localization and Mapping) algorithm based on an Extended Kalman Filter (EKF). This approach decreases the computational time of the correction stage which allows for real and constant-time implementations of the SLAM. The selection procedure consists in choosing the features the SLAM system state covariance is more sensible to. The entire system is implemented on a mobile robot equipped with a range sensor laser. The features extracted from the environment correspond to lines and corners. Experimental results of the real time SLAM algorithm and an analysis of the processing-time consumed by the SLAM with the feature selection procedure proposed are shown. A comparison between the feature selection approach proposed and the classical sequential EKF-SLAM along with an entropy feature selection approach is also performed.

**Keywords:** SLAM, Mobile Robots, Feature Selection

## 1. Introduction

The mobile robotic research area can be divided into three main lines: Mapping, Localization and Navigation /Exploration. Mapping is related to the way the robot models the environment based on the sensors data (Montemerlo, M. & Thrun, S., 2007; Hähnel, D. et al., 2003). Localization is the knowledge of the robot's position within the environment through which it is moving or acting with (Guivant, J. E. & Nebot, E. M., 2001; Xi, B. et al., 2008). Navigation (or exploration) is the strategy used to drive the mobile robot motion in the environment (Arkin, R. C., 1998; Thrun, S. et al., 2005).

In order to reach full autonomy, the robot should be capable of localizing itself in the environment through which is navigating while building a map of such an environment, without human intervention on the mobile robot's motion commands (Montemerlo, M. & Thrun, S., 2007; Thrun, S. et al., 2005). The SLAM (Simultaneous Localization and Mapping) algorithm comes up as a solution to the concurrent tasks of map building and robot localization (Durrant-Whyte, H. & Bailey, T., 2006; Durrant-Whyte, H. & Bailey, T., 2006a).

From its early beginning, SLAM has been implemented in several solution algorithms, depending on the application of the mobile robot task. The Extended Kalman Filter (EKF) solution is the most widely approach used in the scientific community (Ayache, N. & Faugeras, O., 1989; Chatila, R. & Laumond, J. P., 1985), though it is restricted to modeling every variable as a Gaussian random variable. The Particle Filter (PF) and the Unscented Kalman Filter (UKF) have proven to be better approaches to the SLAM problem. The Particle Filter solves the Gaussian restriction of the variables involved in the

SLAM models (Thrun, S. et al., 2005) whereas the UKF has shown a better performance dealing with non-linear models of the vehicle and the measurements (Thrun, S. et al., 2005). Nevertheless, both algorithms are computer-time heavy.

The model of the environment built by the SLAM algorithm represents the way the robot senses and recognizes the environment, depending on the sensor type mounted on the vehicle. The map can be either topological, metrical or hybrid (Ayache, N. & Faugeras, O., 1989; Kouzoubov, K. & Austin, D., 2004), being the metric feature-based map the most widely used technique (Andrade-Cetto, J. & Sanfeliu, A., 2002). The extracted features usually consist on some geometrical constrain of the environment, such as lines, corners or curves. These features are used to localize the robot within the environment.

Since the SLAM algorithm is a dimensionally dynamic algorithm, its complexity grows as the number of observed features increases. Thus, the sampled time of the entire algorithm is variable. This situation is a hindrance when the SLAM is combined with some planned-based control strategy. The reason for this is that, during the SLAM process, the vehicle keeps operating in an open loop control. Some approaches have been proposed to solve this problem (Castellanos, J. et al., 2007) for real time implementation, usually sacrificing the map's resolution (Castellanos, J. et al., 2007; Guivant, J. E. & Nebot, E. M., 2001) or, else, they may lack a plan-based control strategy (Andrade-Cetto, J. & Sanfeliu, A., 2002).

Some approaches concerning a feature selection restriction in the SLAM algorithm can be found in the scientific literature. In (Zhang, D. et al., 2005) a restriction based on the entropy analysis of the feature is made

before updating the SLAM system state. This way, if the entropy is higher than a certain threshold, then it is considered for updating the system state. In the visual SLAM presented in (Fintrop, S. et al., 2006), the algorithm detects regions with strong contrast while rejecting the remaining areas. In (Eade, E. & Drummond, T.W., 2006), a monocular SLAM based on edge detection is presented. The edges are selected according to the order of their geometrical information. Thus, the edges with higher order of geometrical information are used in the SLAM process. In (Lee, Y.J. & Song, J.B., 2007), a visual SLAM algorithm based on object recognition is proposed. In this paper, a certain number of objects are recognized, classified and then used in the SLAM process. Excepting the paper by (Zhang, D. et al, 2005), the remaining works in the literature are focused on selecting the features that better represent the environment, or those which are better detected, though without showing how the selected features will influence the SLAM process.

Our paper proposes a feature selection procedure based on SLAM posterior covariance sensibility. This method recursively selects those features that will make the EKFSLAM to converge faster. The determinant of the SLAM covariance matrix is used as a measurement of the volume of the uncertainty ellipse associated with the SLAM system state. Restricting the EKF correction stage to the selected features allows for real and constant time implementation without losing map consistence. The extracted features correspond to lines and corners, either concave or convex.

The paper is organized as follows. Section 2 shows an introduction to the proposed SLAM algorithm, and two feature selection SLAM approaches. One is based on the entropy of the extracted features, while the other approach —representing the contribution of this paper— is based on the sensibility of the covariance matrix of the SLAM system state. Section 3 presents the comparison results between the two feature selection SLAMs and the experimental results with the entire system implemented. Finally, section 4 shows some conclusions of this work.

## 2. Methods

In this section, the general EKF-SLAM used here is presented, along with two procedures to select the features of the SLAM correction stage. A first approach is based on the entropy analysis of the environment features, and the other one is based on the sensibility of the covariance matrix of the SLAM system state.

### 2.1. EKF-SLAM Algorithm

The SLAM algorithm solved by an EKF is stated by Eqs. (1) - (5). All variables involved in the estimation process are considered as Gaussian random variables.

Prediction Stage

$$\hat{\xi}_t^- = f(\hat{\xi}_{t-1}^-, u_t) \quad (1)$$

$$P_t^- = A_t P_{t-1} A_t^T + W_t Q_{t-1} W_t^T \quad (2)$$

Correction Stage

$$K_t = P_t^- H_t^T (H_t P_t^- H_t^T + R_t)^{-1} \quad (3)$$

$$\hat{\xi}_t = \hat{\xi}_t^- + K_t (z_t - h(\hat{\xi}_t^-)) \quad (4)$$

$$P_t = (I - K_t H_t) P_t^- \quad (5)$$

In (1) - (2) the expression for the prediction stage of the EKF is presented.  $\hat{\xi}_t^-$  is the predicted state of the system at time  $t$ ;  $u_t$  is the input control commands and  $\hat{\xi}_t$  is the corrected state at time  $t$ ;  $f$  describes the motion of the elements of  $\hat{\xi}$ .  $P_t^-$  and  $P_t$  are the predicted and corrected covariance matrices respectively, at time  $t$ ;  $A_t$  is the Jacobian of  $f$  with respect to the SLAM system state and  $Q_t$  is the covariance matrix of the noise associated to the process, whereas  $W_t$  is its Jacobian matrix. Equations (3) - (5) represent the correction stage of the EKF-SLAM.  $K_t$  is the Kalman gain of the algorithm at time  $t$ ;  $H_t$  is the Jacobian matrix of the measurement model ( $h$  in (4)) and  $R_t$  is the covariance matrix of the actual measurement  $z_t$ . In (4),  $(z_t - h(\hat{\xi}_t^-))$  is called the innovation vector (Thrun, S. et al., 2005) and takes place when the data association procedure has reached an appropriate matching between the observed feature and the predicted one ( $h(\hat{\xi}_t^-)$ ). Both, the process model ( $f$  in (1)) and the observation model are non-linear expressions. In this work, the features extracted from the environment correspond to lines and corners, either concave or convex.

The SLAM system state is composed by both the vehicle's degrees of freedom and the parameters that define all features, considered as stationary. According to this, the motion computed in (1) corresponds to the robot's movements. More information about this topic can be found in (Durrant-Whyte, H. & Bailey, T., 2006; Durrant-Whyte, H. & Bailey, T., 2006a). The mobile robot used in this work is a non-holonomic unicycle type -Pioneer 3DX, built by ActivMedia®-. Figure 1 shows a scheme of the vehicle and (6) represents their kinematic equations. The robot has a range sensor laser -built by SICK®- which acquires 181 measurements, from 0 to 180 degrees, within a 30-meter range. The models of the features are shown in equations (7) - (8), and Fig. 2 shows the geometrical meanings of each variable used in (7) and (8). More information about features extraction and the SLAM algorithm implemented in this work can be found in (Garulli, A. et al., 2005; Andrade-Cetto, J. & Sanfeliu, A., 2002).

$$\begin{pmatrix} x_t \\ y_t \\ \Psi_t \end{pmatrix}_{Global} = \begin{pmatrix} x_{t-1} \\ y_{t-1} \\ \Psi_{t-1} \end{pmatrix} + \Delta t \begin{pmatrix} \cos(\Psi_{t-1}) & -a \sin(\Psi_{t-1}) \\ \sin(\Psi_{t-1}) & a \cos(\Psi_{t-1}) \\ 0 & 1 \end{pmatrix} \begin{pmatrix} u_t \\ \omega_t \end{pmatrix} + \Phi_t \quad (6)$$

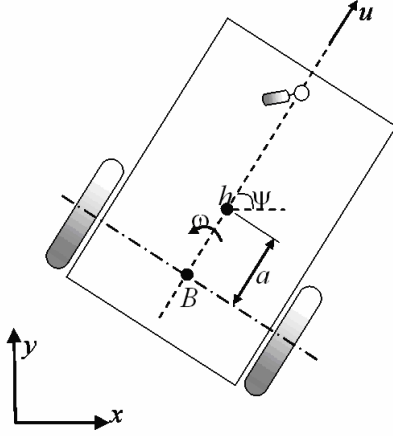


Fig. 1. Schematic of the mobile robot used.

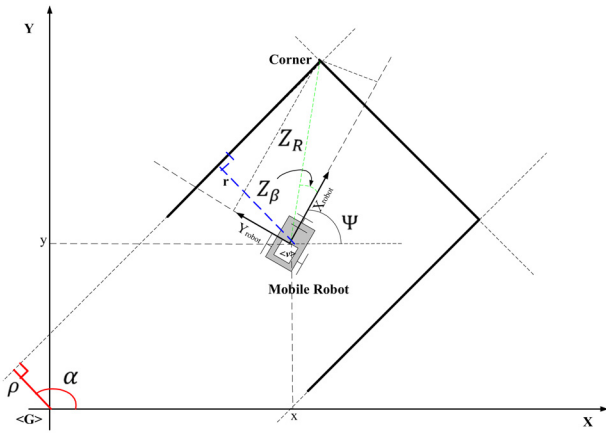


Fig. 2. Feature extraction.

$$z_{corner_t} = h_t[Pose_{robot_t}, p_{i_t}, w_t] = \begin{bmatrix} z_R \\ z_\beta \end{bmatrix} = \begin{bmatrix} \sqrt{(x_t - x_e)^2 + (y_t - y_e)^2} \\ \text{atan}\left(\frac{x_t - y_e}{x_t - x_e}\right) - \Psi_t \end{bmatrix} + \begin{bmatrix} w_R \\ w_\beta \end{bmatrix} \quad (7)$$

$$z_{line_t} = h_t[Pose_{robot_t}, p_{i_t}, w_t] = \begin{bmatrix} z_\rho \\ z_\alpha \end{bmatrix} = \begin{bmatrix} r - x_t \cos(\alpha) - y_t \sin(\alpha) \\ \alpha - \Psi_t \end{bmatrix} + \begin{bmatrix} w_\rho \\ w_\alpha \end{bmatrix} \quad (8)$$

In (6),  $\Phi$  is the discrete time Gaussian process noise associated with the robot's model,  $u$  is the linear velocity command applied to the vehicle and  $\omega$  is the angular velocity command. In (7) and (8),  $w$  signals  $w_R, w_\beta, w_\rho, w_\alpha$  represent the Gaussian measurement noise associated with the observation model.

In this work, a sequential EKF algorithm was implemented as in (Thrun, S. et al., 2005). In this algorithm, the correction iteration is executed with one innovation at a time. This algorithm implies that the Jacobian matrix of the observation model,  $H$ , be a sparse matrix as shown in (Thrun, S. et al., 2005).

## 2.2. Sequential EKF-SLAM: General Algorithm

The general form of the sequential EKF-SLAM algorithm (Thrun, S. et al., 2005) is summarized in Algorithm 1. Sentences (iii) to (x) describe *for-loop* of the correction stage of the algorithm. For every feature with correct association -(ii)- the *for-loop* is executed. In the next iteration, the actual predicted covariance is the last corrected SLAM system state covariance matrix, as noted in sentence (ix).

Let  $N_t$  be the set of observed features (i)

Let  $M_t \subseteq N_t$  be the set of features with correct matching (ii)

for  $i = 1 : \#M_t$  (iii)

$$K_t = P_t^- H_t^T (H_t P_t^- H_t^T + R_t)^{-1} \quad (iv)$$

$$\hat{\xi}_t = \hat{\xi}_t^- + K_t (z_t - h(\hat{\xi}_t^-)) \quad (vi)$$

$$P_t = (I - K_t H_t) P_t^- \quad (vii)$$

$$M_t = M_t - \{z_t^{opt}\} \quad (viii)$$

$$P_t^- := P_t \quad (ix)$$

end for (x)

Algorithm 1. Sequential EKF-SLAM without Feature Selection.

## 2.2. SLAM with feature selection: Entropy Approach

The SLAM algorithm with feature selection based on the observation of the entropy of the measurements was previously presented by (Zhang, D. et al., 2005). This algorithm is considered as related to the one proposed in this paper and will be used for comparison purposes. This method is based on the calculation of the entropy attached to each observed feature. If the entropy is below a certain threshold value, then the observation will be computed in the correction stage of the EKF.

Considering that all variables involved in the EKF-SLAM estimation process are Gaussian random variables, the entropy value associated with a single observation can be represented as in (9).

$$\Sigma_t = E\{-\ln p(\xi_t | z_t)\} = 0.5 \ln \left[ (2\pi e)^n |P_t| \right] \quad (9)$$

Here  $\Sigma$  is the entropy of the observation  $z$ . The *a priori* and *posteriori* information metric can be defined as the inverse of the entropy value, shown in (9). Thus,

$$im_t = -\Sigma_t = -0.5 \ln \left[ (2\pi e)^n |P_t| \right] \quad (10)$$

$$im_t^- = -\Sigma_t^- = -0.5 \ln \left[ (2\pi e)^n |P_t^-| \right] \quad (11)$$

The information difference can be calculated as in (12), where the absolute incremental information is obtained.

$$\Delta_i = im_t - im_t^- \quad (12)$$

Thus, when the absolute information of a feature exceeds a certain threshold ( $\delta$ ), that feature will be used in the correction stage of the EKF-SLAM algorithm. The complete algorithm can be summarized as follows.

Let  $N_t$  be the set of all  $z$  at time  $t$

Let  $M_t \subseteq N_t$  be the set of all  $z$  with appropriate matching at time  $t$

$\forall z_i \in M_t, P_t \mid z_i$  is calculated according to (5)

If  $\Delta_i \geq \delta$ , then the update (Eq. (4)) takes place.

Algorithm 2. Entropy feature selection procedure.

As Algorithm 2 shows, the calculation of the entropy associated with a single observation -and its information metric- is related to the determinant of the complete covariance matrix of the SLAM system state. Thus, the complexity of the calculation of the entropy is  $O(n^2)$ , where  $n$  is the dimension of the SLAM system state. Since this dimension varies, the complexity of the algorithm varies as well.

Although this algorithm has the advantage of restricting the number of features to be updated, the benefits in processing time are shown only in the updating part of the EKF-SLAM (Eq. (4)). The reason for this is that the calculation of the entropy requires a previous calculation of both the covariance matrix (Eq. (5)) and its determinant which, with a higher number of states, will delay the execution of the EKF-SLAM algorithm. Further detail on this approach can be found in (Zhang, D. et al, 2005).

### 2.3. SLAM with Feature Selection: Covariance Sensibility Approach

The proposal of this work is a feature selection algorithm based on the observation of the sensibility of the SLAM covariance matrix. This approach is sustained by evaluating the influence of a given feature of the environment -with positive matching in (4)- in the convergence ratio of the covariance matrix of the SLAM system state. Thus, according to (5), the covariance correction can be expressed as:

$$P_t = (I - K_t H_t) P_t^- = P_t^- - K_t H_t P_t^-.$$

By applying the determinant to both sides of the equality above,

$$|P_t| = |(I - K_t H_t)| |P_t^-| = |P_t^- - K_t H_t P_t^-| \quad (13)$$

and considering that, in (13)  $P_t^-$  and  $K_t H_t P_t^-$  are *psd* (positive semi-definite) matrices, the above expression leads to (14).

$$|P_t| = |(I - K_t H_t)| |P_t^-| = |P_t^- - K_t H_t P_t^-| \leq |P_t^-| \quad (14)$$

Then, from (13) and (14),

$$\begin{cases} 0 \leq \frac{|P_t|}{|P_t^-|} \leq 1, \text{ with } |P_t^-| \neq 0 \\ 0 \leq \frac{|P_t|}{|P_t^-|} = |I - K_t H_t| \leq 1, \end{cases} \quad (15)$$

Equation set (15) implicitly defines the convergence property of the EKF-SLAM, which is: *the determinant of the correction covariance is always smaller than or equal to the determinant of the predicted covariance*. The determinant of the covariance matrix, in this case, is used as a measure of the volume of the uncertainty ellipse associated with that matrix (Castellanos, J. et al., 2004).

Another convergence property states that, at the limit, all elements of  $P_t$  become fully correlated (Castellanos, J. et al., 2004). This last statement is equal to say that,

$$\lim_{t \rightarrow \infty} |P_t| = 0. \quad (16)$$

Statement (15) defines the sensibility of the covariance matrix of the SLAM system state measured from the determinant of such a matrix. Thus, according to (15) and (16), given a set of observed features with correct matching, the feature that causes the highest decrease of  $|P_t|$ , is the feature to which the EKF-SLAM is more sensitive to and will cause the fastest convergence of (16). This latter point can be regarded as an optimization problem. Let  $N_t$  be the set of observed features at time  $t$ ; let  $M_t \subseteq N_t$  be the set of features with correct matching. Then,  $\forall z \in M_t \subseteq N_t$ :

$$z^{opt} : \arg \min_z (|P_t|) \equiv \arg \min_z (|I - K_t H_t|). \quad (17)$$

Thus, according to (17), finding the observation  $z$  that minimizes  $|I - K_t H_t|$  is equivalent to finding the observed feature that causes the highest decrease of  $|P_t|$ . This is so because  $|P_t^-|$  is independent of the current observation (see (2)).

Considering now that the EKF-SLAM implemented in this work is a sequential algorithm (Thrun, S. et al., 2005), the Jacobian of the observation model has the form shown in (18), where  $H_{v,t}$  is the Jacobian of the observation model with respect to the vehicle's variables and  $H_{z,t}$  is the Jacobian of the observation model with respect to the parameters of the observed feature. The Kalman gain -(19)- is also defined according to (18).

$$H_t = \begin{bmatrix} H_{v,t} & \Theta_1 & H_{z,t} & \Theta_2 \end{bmatrix} \quad (18)$$

$$K_t = \begin{bmatrix} K_{v,t}^T & K_{\Theta 1}^T & K_{z,t}^T & K_{\Theta 2}^T \end{bmatrix}^T \quad (19)$$

In (18),  $\Theta$  is a null matrix. Thus, the Jacobian of the observation is only calculated on the Jacobian entries that correspond to the vehicle and to the feature with correct matching. More information about this topic can be found in (Durrant-Whyte, H. & Bailey, T., 2006; Durrant-Whyte, H. & Bailey, T., 2006a). Using (18) and (19), the expression  $|I - K_t H_t|$  can be calculated as:

$$\begin{aligned}
 |I - K_t H_t| &= \left| \begin{bmatrix} I_v & & & \\ & I_{\Theta 1} & & \\ & & I_z & \\ & & & I_{\Theta 2} \end{bmatrix} - \begin{bmatrix} K_{v,t} \\ K_{\Theta 1} \\ K_{z,t} \\ K_{\Theta 2} \end{bmatrix} \begin{bmatrix} H_{v,t} & \Theta_1 & H_{z,t} & \Theta_2 \end{bmatrix} \right| \\
 &= \begin{vmatrix} I_v - K_{v,t} H_{v,t} & \Theta & -K_{v,t} H_{z,t} & \Theta \\ -K_{\Theta 1} H_{v,t} & I_{\Theta 1} & -K_{\Theta 1} H_{z,t} & \Theta \\ -K_{z,t} H_{v,t} & \Theta & I_z - K_{z,t} H_{z,t} & \Theta \\ -K_{\Theta 2} H_{v,t} & \Theta & -K_{\Theta 2} H_{z,t} & I_{\Theta 2} \end{vmatrix} \\
 &= \begin{vmatrix} I_v - K_{v,t} H_{v,t} & -K_{v,t} H_{z,t} \\ -K_{z,t} H_{v,t} & I_z - K_{z,t} H_{z,t} \end{vmatrix} \quad (20)
 \end{aligned}$$

In (20),  $I$  is the identity matrix,  $I_v$ ,  $I_{\Theta 1}$ ,  $I_z$  and  $I_{\Theta 2}$  are identity block matrices with the dimensions of  $K_{v,t} H_{v,t}$ ,  $K_{\Theta 1} \Theta_1$ ,  $K_{z,t} H_{z,t}$  and  $K_{\Theta 2} \Theta_2$  respectively. Thus, considering that the vehicle has three degrees of freedom and a feature -corner or line- is determined within two parameters -see (7) and (8)-, the final calculation in (20) is a determinant of a  $5 \times 5$  matrix.

The final correction stage algorithm implemented in this work is formalized in Algorithm 3.

Let  $N_t$  be the set of observed features (i)

Let  $M_t \subseteq N_t$  be the set of features with correct matching (ii)

Let  $LIM$  be the maximum number of features to be used in the correction stage (iii)

for  $i = 1 : \min\{LIM, \#M_t\}$  (iv)

find  $z_i^{opt} : \arg_z \min(|P_t|) \equiv \arg_z \min(|I - K_t H_t|)$  (v)

$K_t = P_t^{-T} H_t^T (H_t P_t^{-T} H_t^T + R_t)^{-1} z_i^{opt}$  (vi)

$\hat{\xi}_t = \hat{\xi}_t^- + K_t (z_i^{opt} - h(\hat{\xi}_t^-))$  (vii)

$P_t = (I - K_t H_t) P_t^- | z_i^{opt}$  (viii)

$M_t = M_t - \{z_i^{opt}\}$  (ix)

$P_t^- := P_t$  (x)

end for (xi)

Algorithm 3. Feature Selection based on Covariance Sensibility

In Algorithm 3, sentences (i) - (ii) are the declaration of the domain that is going to be used in the correction stage; sentence (iii) determines -if possible- the maximum number of features that will be used for correcting the SLAM. If the number of features in  $M_t$  is smaller than  $LIM$ , then the complete set of features in  $M_t$  will be used in the correction loop. Sentences (iv) - (xi) show the *for*-

loop of the correction stage. Given  $M_t$ , the algorithm searches for a first  $z^{opt}$ . When it is found, the correction stage takes place -(vi) to (viii)- and this feature is removed from  $M_t$ . In the second iteration of the *for*-loop, the  $z^{opt}$  is searched inside the new  $M_t$  and the actual predicted covariance is the last corrected system state covariance matrix as shown in sentence (x). This situation ensures that sequentiality of the EKF-SLAM is not lost.

Comparing Algorithm 2 with Algorithm 3 it is possible to see that Algorithm 3 does not compute the update of the SLAM covariance system state until a  $z^{opt}$  is found and only for a limited number of  $z^{opt}$ . On the other hand, Algorithm 2 must calculate the correction of the covariance matrix in order to obtain the entropy value. This situation requires the system to execute  $O(n^2)$  iterations for every feature -besides the  $O(n^2)$  computational cost required for the calculation of the determinant of  $P_t^-$ .

Let  $n$  be the number of features with correct matching in Algorithm 1 ( $\#M_t = n$ ). Then, the determinant of the final system state covariance matrix can be calculated as

$$\begin{aligned}
 |P_t^-|_{z_1} &= |P_t| = |I - K_t H_t|_{z_1} |P_t^-| \\
 |P_t^-|_{z_2} &= |P_t| = |I - K_t H_t|_{z_2} |P_t^-|_{z_1} \\
 &\dots \\
 \Rightarrow |P_t| &= |I - K_t H_t|_{z_1} \times \dots \times |I - K_t H_t|_{z_n} |P_t^-|. \quad (21)
 \end{aligned}$$

In addition, in Algorithm 3, let us suppose that the maximum number of features to be used in the correction stage is  $LIM = 2$  and that  $\#M_t > 2$ . Then, the determinant of the final system state covariance matrix can be calculated as

$$\begin{aligned}
 |P_t^-|_{z_1} &= |P_t| = |I - K_t H_t|_{z_1} |P_t^-| \\
 |P_t| &= |I - K_t H_t|_{z_2} |P_t^-|_{z_1} \\
 \Rightarrow |P_t| &= |I - K_t H_t|_{z_1} \times |I - K_t H_t|_{z_2} |P_t^-|. \quad (22)
 \end{aligned}$$

Then, considering that  $|I - K_t H_t|_{z_i} \leq 1$ ,  $i = 1, \dots, n$ , from Eqs. (21) and (22),

$$|P_t|_{\text{Algorithm1 Sequential EKF-SLAM}} \leq |P_t|_{\text{Algorithm3}}. \quad (23)$$

Then, from Eqs. (21) to (23), the following expression holds,

$$\frac{|P_t|_{\text{Algorithm1}}}{|P_t|_{\text{Algorithm3}}} \leq |I - K_t H_t|_{z_3} \times \dots \times |I - K_t H_t|_{z_n} \leq 1. \quad (24)$$

Equation (24) can be used as a measurement means for the accuracy of the system state estimation using the EKF-SLAM with feature selection based on covariance sensibility, with respect to the classical sequential EKFSLAM. Then, the higher the ratio shown in (24), the better the system state estimation made by the EKF-SLAM with the feature selection procedure proposed in this work.

### 3. Implementation results

Several experiments were carried out in order to show the performance and benefits of the EKF-SLAM with feature selection criteria based on the covariance matrix sensibility proposed in this work. The computational cost of this algorithm was also compared to the sequential EKF-SLAM and the entropy approach EKF-SLAM shown in sections 2.2 and 2.3 respectively. All SLAM algorithms worked in real time. The mobile robot motion strategy used in the experiences was a local navigation based on frontier points determination (Tao, T. et al., 2007). Considering that the SLAM provides an estimate of the robot's pose to the control algorithm, only one SLAM algorithm can be executed at a time. The following sections show the adopted control strategy in detail, the SLAM consistence analysis and the computational costs of each SLAM algorithm.

#### 3.1. Mobile Robot Motion Strategy

The motion strategy used in this work obeys the concept of Active SLAM introduced by (Tao, T. et al., 2007). Thus, the robot's pose estimation is used to generate the controller references in order to drive the robot. In this work, a navigation based on frontier points was implemented, which can be summarized as follows.

- The robot first searches for an unoccupied space at the frontier of its range sensor measurements (frontier points).
- Once a frontier point is selected, a straight trajectory is generated from the robot's position to the frontier point.
- A trajectory controller is then executed to reach a neighborhood of the frontier point. This controller is an hybrid dynamic adaptive controller presented in (De la Cruz, C. et al., 2008).
- Once the robot reaches a neighborhood of the destination point, a next frontier point is searched. If no frontier point is detected, then the robot turns around until it finds a frontier point.
- If during the trajectory an obstacle is detected, the robot avoids it using a tangential avoidance criteria (Santos Brandao, A. et al., 2008) without losing its destination objective.
- The SLAM algorithm keeps being active during the control execution.

Figure 3 shows the Active SLAM architecture implemented in this work. The localization and mapping procedures are carried out at the global reference frame, i.e., the system state variables are defined at this global frame. Nevertheless, the EKF-SLAM is executed at a local reference frame level. Thus, the data association and the features' prediction and extraction are taken into account at the local reference frame attached to the robot. The frontier points detection, the trajectory planning and the trajectory control are calculated and executed at the local reference frame with the information stemming from sensor measurements.

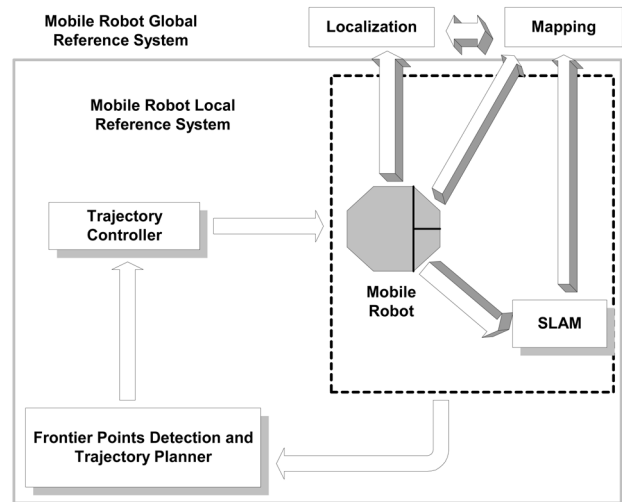


Fig. 3. Active SLAM architecture. The mobile robot navigation performs at a local reference frame attached to the robot.

The mobile robot used in this work, is an unicycle type, Pioneer 3DX built by ActivMedia® with a range laser sensor built by SICK®. The laser takes 181 measurements from the environment, from 0 - 180°. A maximum range of 10 meters was adopted in this work.

#### 3.2. Consistence Analysis

Although the EKF-SLAM with the feature selection criteria proposed in this work does not imply a modification of the EKF, a consistence analysis was carried out for this SLAM. The consistence analysis shows that, when the EKF correction stage is restricted to those features that contribute the most for SLAM system state covariance matrix convergence (see Eq. (14)), the map of the environment does not become inconsistent.

In order to show consistence, the true location of the robot in the environment is needed for comparison purposes. Figure 4.a shows the simulated environment with the expected trajectory. The environment was created using the MobileSim® tools (provided by ActivMedia®). Figure 4.b shows the map reconstruction and the actual path travelled by the robot. The mobile robot drives according to the control strategy presented in the previous section. The green circles (o) correspond to corners of the environment; the blue points represent the travelled path by the mobile robot; the red crosses (x) are the beginning and ending points of the segments associated with the lines of the environment. Although the SLAM algorithm presented in this work has the parameters of lines and corners in its system state, the beginning and ending points of each line are stored in a secondary map. This map is updated according the SLAM system state updating. Finally, the solid black lines in Fig. 4.b are the segments associated to walls of the environment. Figures 4.c and 4.d show that the robot's position error remains bounded by  $2\sigma$  (two times the standard deviation).



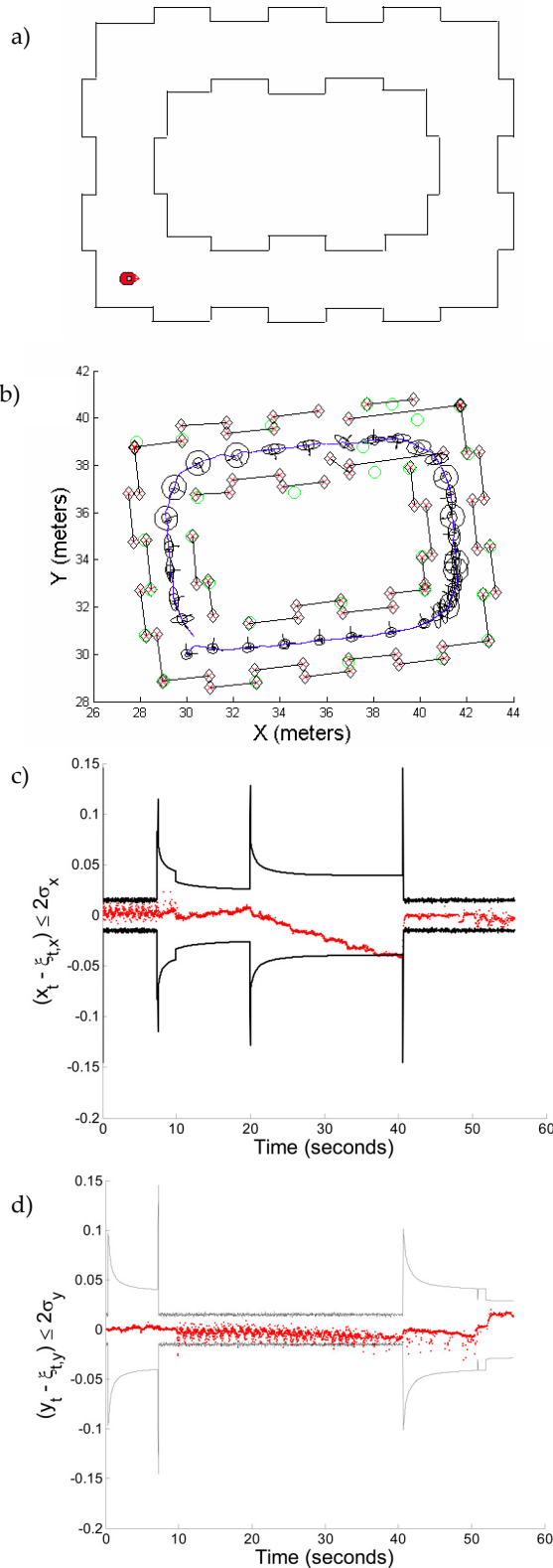


Fig. 4. Consistence analysis of the EKF-SLAM with feature selection criteria based on SLAM system state covariance matrix sensibility. a) Simulated environment with expected path; b) map reconstruction based on the SLAM system state. It can be seen that the map is coherent with its model; c) the  $x$ -robot position in the environment is bounded by two times its variance; d) idem for  $y$ -robot position.

### 3.3. Experimental Results

According to section 3.1, the mobile robot trajectory controller uses the pose estimation of the mobile robot for generating the motion references. This scenario implies that only one SLAM algorithm is able to perform its estimation at a time. Therefore, three similar experiments were made in the facilities of the Instituto de Automatica. For each experience, the robot was placed in the same initial position in the building. Figure 5.a-c shows the map reconstruction of the places traversed of the Instituto de Automatica according to the three EKFSLAM considered in this paper (sequential EKF-SLAM, entropy feature selection approach and the covariance matrix sensibility approach). In Fig. 5, the green circles represent corners of the environment, the blue points are the path travelled by the robot, the red crosses are the beginning and ending points of the segments of the environment, whereas solid black lines are the segmentes associated with lines and the yellow points are raw laser data.

The accumulated processing time, as can be seen in Fig. 6, shows that the EKF-SLAM with feature selection based on the SLAM system state with the covariance sensibility approach has shorter computational processing time when compared with the sequential EKF-SLAM and the entropy feature selection approach. The accumulated processing time shown in Fig. 6 corresponds to the experiments of Fig. 5. After several long-time trials, the accumulated time of the EKF-SLAM with covariance sensibility approach was 33% shorter than the accumulative time of the sequential EKF-SLAM and 47% shorter than the EKF-SLAM with entropy approach. In addition, the accuracy of the map -see Eq. (24)- for the experiment shown in Fig. 5 was 97.2%.

During the SLAM execution, the mobile robot remains in an open loop state. This period depends on the SLAM processing charges, i.e., on the number of features to be corrected (Thrun, S. et al., 2005). Considering that the sampling time of the mobile robot used in this work (Pioneer 3DX) must be a multiple of 0.1 seconds, if the SLAM requires, e. g., 0.45 seconds of processing, then the sampling time for this iteration will be at least 0.5 seconds. During this time of 0.5 second, the command control of the robot does not change, thus being unable to react if an obstacle appears on its path. This crucial navigation problem can be solved in part by adopting a constant time SLAM. In the sequential EKF-SLAM, a constant processing time is not feasible due to the variation of the number of features to be associated. On the other hand, the EKF-SLAM with feature selection based on SLAM system state covariance matrix sensibility is a constant time algorithm due to the restriction on the number of features to be associated. Figure 7 shows the open-loop accumulated time for the three experiments shown in Fig. 5. As noted, the EKFSLAM with feature selection based on covariance sensibility proposed in this work presents the shortest robot's open loop time.

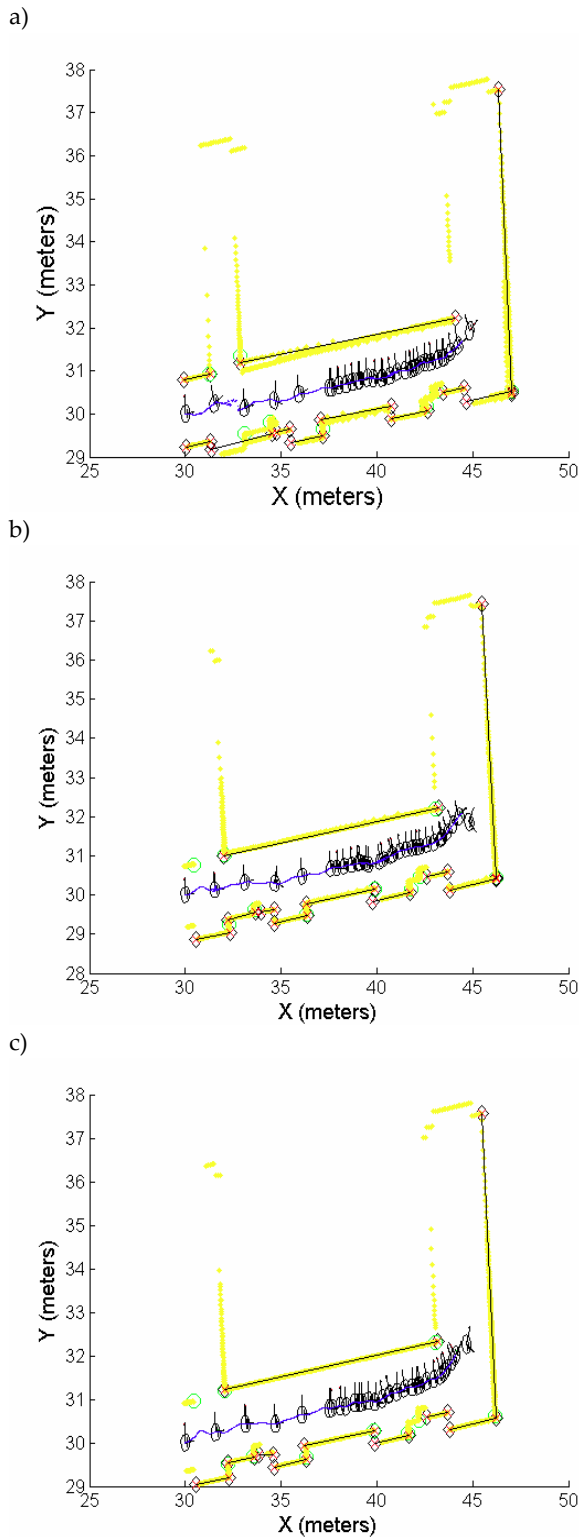


Fig. 5. Real time experimental results. a) Map reconstruction of the facilities at the Instituto de Automatica using the sequential EKF-SLAM system state; b) map reconstruction using the EKF-SLAM with entropy feature selection system state, and c) map reconstruction using the EKF-SLAM with covariance sensibility feature selection procedure system state. In the three experiments the robot was left at the same position inside the Instituto de Automatica, although its orientation varied.

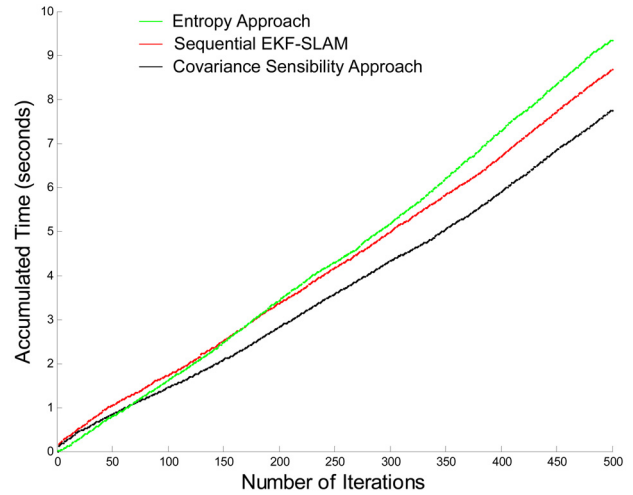


Fig. 6. Accumulated SLAM processing time. The EKFSLAM with the covariance sensibility approach demands lesser processing time per iteration than the classical sequential EKF-SLAM or the entropy feature selection approach.

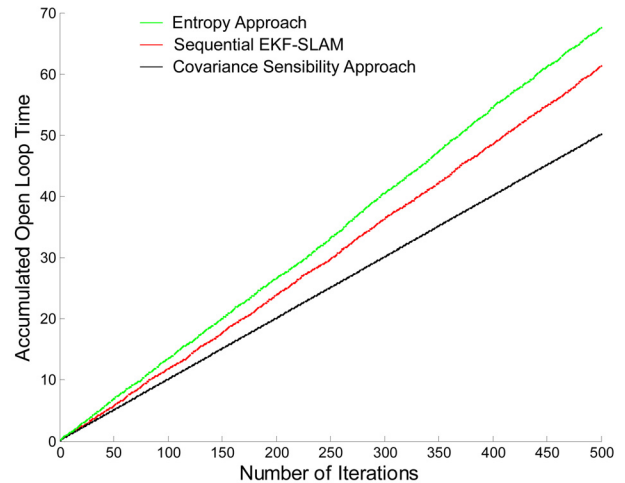


Fig. 7. Accumulated robot's open loop time. The time the robot remains in an open loop state is shorter for the covariance sensibility approach than for the classical sequential EKF-SLAM or for the entropy feature selection approach.

After several trials, the EKF-SLAM with feature selection based on covariance sensibility have shown a 45% shorter accumulated open loop time than the sequential EKF-SLAM, and 56% shorter time than the EKF-SLAM with feature selection based on the feature selection entropy approach.

Since the experiments concerning the EKF-SLAM with feature selection based on the covariance matrix shown in this work had a restricted number of correction iterations (see Algorithm 3), the mobile robot sampling time turned to be 0.2 seconds. For the sequential EKF-SLAM it rose to 0.5 seconds (its maximum registered value), and for the EKF-SLAM with feature selection based on the entropy approach it had a maximum sampling time of 0.7 seconds.



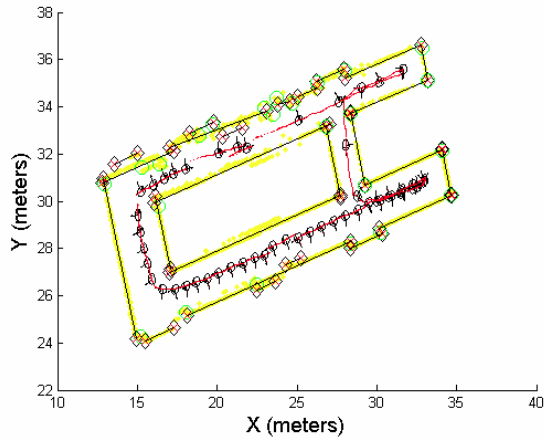


Fig. 8. Map reconstruction of the Instituto de Automatica using the proposed feature selection procedure. Green circles (o) represent corners; red crosses (x) are the beginning and ending points of the segments associated with lines; solid black lines represent walls and yellow points are raw laser data.

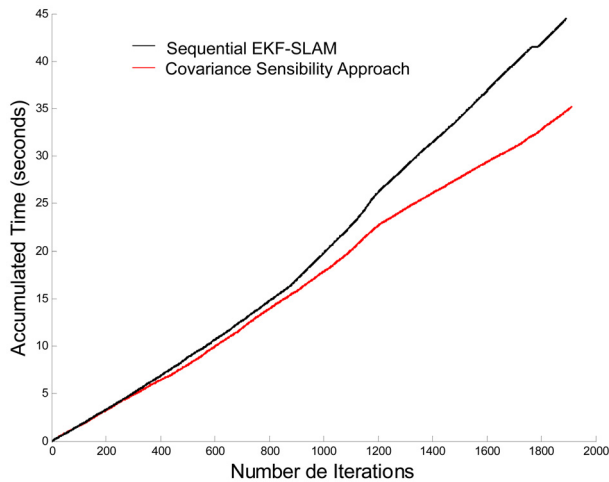


Fig. 9. Accumulated processing time of the EKF-SLAM with feature selection based on covariance sensibility approach when compared to the classical sequential EKF-SLAM. Both experiments consisted on navigating all the corridors within the Instituto de Automatica. In both cases, the robot had performed similar paths.

Figure 8 shows the map reconstruction of the entire Instituto de Automatica when the robot navigates within its premises using the control strategy presented in this work along with the EKF-SLAM with feature selection based on SLAM system state covariance matrix sensibility. Figure 9 shows the accumulated time when a similar navigation is performed using the classical sequential EKF-SLAM.

#### 4. Conclusions

A new solution to the sequential EKF-SLAM algorithm based on a feature selection procedure has been proposed in this work. The feature selection method consisted in correcting the EKF according to those features to which

the SLAM covariance matrix shows more sensibility, allowing, in that way, EKF convergence. The experiments were made in real time realizations and within a real semi-structured environment. A comparison was also made between the approach presented in this work and a feature selection procedure based on entropy analysis along with the classical sequential EKF-SLAM algorithm. The processing times of the different realizations have shown the advantages of using the approach presented in this paper with respect to the entropy analysis and the classical sequential EKF-SLAM. The EKF-SLAM with feature selection based on the covariance sensibility approach have shown an average accumulated processing time 45% shorter than the sequential EKFSLAM and 56% shorter than the EKF-SLAM with feature selection based on the entropy approach. The experiments have also shown that the map kept being consistent.

The EKF-SLAM with feature selection based on the covariance sensibility approach has allowed to execute a constant time SLAM in real time realizations.

#### 5. References

- Andrade-Cetto, J. & Sanfeliu, A. (2002). Concurrent Map Building and Localization with Landmark Validation. *International Journal of Pattern Recognition and Artificial Intelligence*. 16(3), pp. 361-374, ISSN: 0218-0014.
- Arkin, R. C. (1998). *Behavior-based Robotics*, ISBN: 0-262-01165-4, MIT Press: Cambridge, MA.
- Ayache, N. & Faugeras, O. (1989). Maintaining a representation of the environment of a mobile robot, *IEEE Transactions on Robotics and Automation*, vol. 5, no. 6, pp. 804-819, ISSN: 1042-296X.
- Castellanos, J.; Martinez-Cantin, R.; Tardos, J. & Neira, J. (2007). Robocentric map joining: Improving the consistency of EKF-SLAM, *Robotics and Autonomous Systems*, vol. 55, pp. 21-29, ISSN: 0921-8890.
- Castellanos, J.; Neira, J. & Tardos, J. (2004). Limits to the consistency of EKF-based SLAM, in 5th IFAC Symposium on Intelligent Autonomous Vehicles, ISBN: 0-08-044333-8, Lisbon, Portugal.
- Chatila, R. & Laumond, J. P. (1985). Position referencing and consistent world modeling for mobile robots, *Proceedings of IEEE International Conference on Robotics and Automation*, pp. 138-145, St. Louis, USA.
- De la Cruz, C.; Carelli, R. & Bastos Filho, T. F. (2008). Switching Adaptive Control of Mobile Robots, *International Symposium on Industrial Electronics*. ISBN : 1-4244-1666-0. UK.
- Durrant-Whyte, H. & Bailey, T. (2006). Simultaneous localization and mapping (SLAM): part I essential algorithms, *IEEE Robotics and Automation Magazine*, Vol 13, No. 2, pp. 99-108, ISSN: 1070-9932.
- Durrant-Whyte, H. & Bailey, T. (2006a). Simultaneous localization and mapping (SLAM): part II state of the art, *IEEE Robotics and Automation Magazine*. Vol 13, No. 3, pp. 108-117, ISSN: 1070-9932.

- Eade, E. & Drummond, T.W. (2006). Edge landmarks in monocular SLAM, in The 17th British Machine Vision Conference (BMVC), August, ISBN: 1-904410-14-6, Edinburgh, UK.
- Fintrop, S.; Jensfelt, P. & Christensen, H. I. (2006). Attentional Landmark Selection for Visual SLAM, in Proc. of the International Conference on Intelligent Robots and Systems, pp. 2582-2587, ISBN: 1-4244-0259-X, Bieijing, China.
- Garulli, A.; Giannitrapani, A.; Rosi, A. & Vicino, A. (2005). Mobile robot SLAM for line-based environment representation, Proceedings of the 44<sup>th</sup> IEEE Conference on Decision and Control, pp. 2041-2046, ISBN: 0-7803-9567-0, December 2005.
- Guivant, J. E. & Nebot, E. M. (2001). Optimization of the simultaneous localization and map-building algorithm for real-time implementation, *IEEE Transactions on Robotics and Automation*, vol. 17, pp. 242-257, ISSN: 0882-4967.
- Hähnel, D.; Fox, D.; Burgard, W. & Thrun, S. (2003). A highly efficient FastSLAM algorithm for generating maps of large-scale cyclic environments from raw laser range measurements, In Proc. of the Conference on Intelligent Robots and Systems (IROS), ISBN: 0-7803-7860-1, October 27-31, Las Vegas, USA.
- Knight, J.; Davison, A. & Reid, I. (2001). Towards constant time slam using postponement, in Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, pp. 406-412, ISBN: 0-7803-6614-X, Maui, Hawaii.
- Lee, Y.J. & Song, J.B. (2007). Autonomous selection, registration and recognition of objects for visual SLAM in indoor environments, in The International Conference on Control, Automation and Systems, October 17-20, ISBN: 978-89-950038-6-2-98560, Seul, Korea.
- Kouzoubov, K. & Austin, D. (2004). Hybrid Topological/Metric Approach to SLAM, Proc. of the IEEE International Conference on Robotics and Automation, ISBN: 0-7803-8232-3, New Orleans, USA.
- Montemerlo, M. & Thrun, S. (2007). *FastSLAM: A factored solution to the simultaneous localization and mapping problem*, ISBN: 978-3-540-46399-3, Springer Tracks in Advanced Robotics. Springer.
- Santos Brandao, A.; Donato, E. B.; Filho, M. S. & Bastos Filho, T. F. (2008). Desvio Tangencial Modificado Aplicado a Navegacao assistida de robos moveis, (Mobile Robot Tangencial Obstacle Avoidance, in Portuguese), Congresso Brasileiro de Automacao. ISBN: 978-85-60750-01-6, Brazil.
- Tao, T.; Huang, Y.; Sun, F. & Wang, T. (2007). Motion Planning for SLAM Based on Frontier Exploration, in Proc. of the IEEE International Conference on Mechatronics and Automation, ISBN: 978-1-4244-0828-3, Takamatsu, Kagawa, Japan.
- Thrun, S.; Burgard, W. & Fox, D. (2005). *Probabilistic Robotics*, ISBN: 978-0-262-20162-9, MIT Press, Cambridge.
- Xi, B.; Guo, R.; Sun, F. & Huang, Y. (2008). Simulation Research for Active Simultaneous Localization and Mapping Based on Extended Kalman Filter. Proc. Of the IEEE International Conference on Automation and Logistics, ISBN: 978-1-4244-2503-7, Qingdao, China.
- Zhang, D.; Xie, L. & David Adams, M. (2005). Entropy based Feature Selection Scheme for Real Time Simultaneous Localization and Map Building, IEEE Conference on Intelligent Robots and Systems, ISBN: 0-7803-8913-1, Edmonton, Canada, August 2-6.