REAL TIME TRACKING OF AN OMNIDIRECTIONAL ROBOT
An Extended Kalman Filter Approach

José Gonçalves, José Lima
Department of Electrical Engineering, Polytechnic Institute of Bragança, Portugal
{goncalves, jllima}@ipb.pt

Paulo Costa
Deec, Faculty of Engineering of the University of Porto, Portugal
paco@fe.up.pt

Keywords: Probabilistic robotics, Kalman Filter, Omnidirectional robots.

Abstract: This paper describes a robust localization system, similar to the used by the teams participating in the Robocup Small size league (SLL). The system, developed in Object Pascal, allows real time localization and control of an autonomous omnidirectional mobile robot. The localization algorithm is done resorting to odometry and global vision data fusion, applying an extended Kalman filter, being this method a standard approach for reducing the error in a least squares sense, using measurements from different sources.

1 INTRODUCTION

Soccer was the original motivation for Robocup. Besides being a very popular sport worldwide, soccer brings up a set of challenges for researchers while attracting people to the event, promoting robotics among students, researchers and general public. RoboCup chose to use soccer game as a central topic of research, aiming at innovations to be applied for socially significant problems and industries (rob, 2008). As robotics soccer is a challenge in an highly dynamic environment, the robot and ball position information must be accessible as fast and accurate as possible (Sousa, 2003). As an example if the ball has a velocity of 2 ms^{-1} and if the lag time is 100 ms, the ball will travel a distance of 20 cm between two sampling instants, compromising the controller performance (Gonçalves et al., 2007). The presented localization algorithm is updated 25 times per second, fulfilling the proposed real time requisites.
Robots maintain a set of hypotheses with regard to their position and the position of different objects around them. The input for updating these beliefs come from poses belief and various sensors (Borestein et al., 1996). An optimal estimation can be applied in order to update their beliefs as accurately as possible. After one action the pose belief is updated based on data collected up to that point in time, by a process called filtering (Thrun et al., 2005).

2 RELATIVE POSITION ESTIMATION

Omnidirectional vehicles are widely used in robotics soccer, allowing movements in every direction, where the extra mobility is an important advantage. The fact that the robot is able to move from one place to another with independent linear and angular velocities contributes to minimize the time to react, the number of maneuvers is reduced and consequently the game strategy can be simplified (Ribeiro et al., 2004). The omnidirectional robots use special wheels, that allow movements in every direction. The movement of these robots does not have the restraints of the differential robots (Dudek and Jenkin, 2000), presenting the disadvantage of a more complex control. It is possible to conclude from the geometry of a three wheel omnidirectional robot, presented in Figure 1, that the velocities V_x, V_y and w vary with the linear velocities V_1, V_2 and V_3, as shown in equations system (1) (Kalmár-Nagy et al., 2002).

$$\begin{pmatrix} V_1 \\ V_2 \\ V_3 \end{pmatrix} = \begin{pmatrix} -sin(\theta) & cos(\theta) & L \\ -sin(\frac{\pi}{3}-\theta) & -cos(\frac{\pi}{3}-\theta) & L \\ sin(\frac{\pi}{3}+\theta) & -cos(\frac{\pi}{3}+\theta) & L \end{pmatrix} \begin{pmatrix} V_x \\ V_y \\ w \end{pmatrix} \tag{1}$$

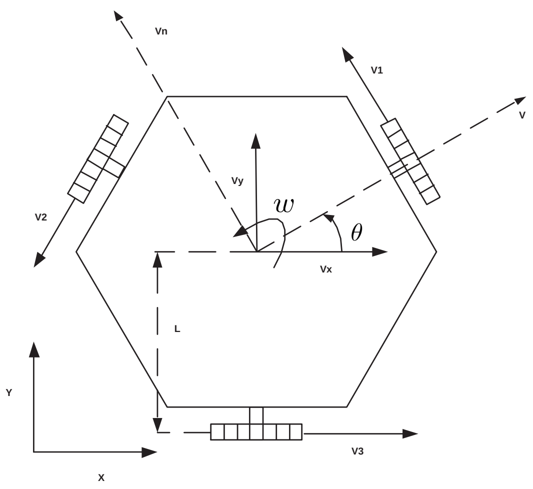

Figure 1: Geometry of a three wheel omnidirectional robot.

2.1 Odometry Calculation

The robot relative position estimation is based on the odometry calculation. The odometry calculation uses each wheel velocity in order to estimate the robot position, the disadvantage is that the position estimate error is cumulative and increases over time.

The robot kinematic equations can be represented by the equations system (2), in alternative to the equations system (1).

$$\begin{pmatrix} V_1 \\ V_2 \\ V_3 \end{pmatrix} = \begin{pmatrix} 0 & 1 & L \\ -sin(\frac{\pi}{3}) & -cos(\frac{\pi}{3}) & L \\ sin(\frac{\pi}{3}) & -cos(\frac{\pi}{3}) & L \end{pmatrix} \begin{pmatrix} V \\ V_n \\ w \end{pmatrix} \quad (2)$$

The linear and angular velocities V, V_n and w can be obtained rewriting equations system (2) as equations system (3),

$$\begin{pmatrix} V \\ V_n \\ w \end{pmatrix} = \begin{pmatrix} G \end{pmatrix} \begin{pmatrix} V_1 \\ V_2 \\ V_3 \end{pmatrix} \quad (3)$$

where G is :

$$\begin{pmatrix} 0 & \frac{-1}{2\sin(\frac{\pi}{3})} & \frac{1}{2\sin(\frac{\pi}{3})} \\ \frac{1}{1+\cos(\frac{\pi}{3})} & \frac{-1}{2(1+\cos(\frac{\pi}{3}))} & \frac{-1}{2(1+\cos(\frac{\pi}{3}))} \\ \frac{\cos\frac{\pi}{3}}{L(1+\cos(\frac{\pi}{3}))} & \frac{1}{2L(1+\cos(\frac{\pi}{3}))} & \frac{1}{2L(1+\cos(\frac{\pi}{3}))} \end{pmatrix} \quad (4)$$

By this way θ can be found, applying an first order approximation, as shown in equation (5),

$$\theta(K) = \theta(K-1) + wT \quad (5)$$

where T is the sampling time.

After θ calculation an rotation matrix, presented in matrix (6), is applied in order to obtain V_x and V_y, as shown in equations system (7),

$$B = \begin{pmatrix} cos(\theta) & -sin(\theta) & 0 \\ sin(\theta) & cos(\theta) & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad (6)$$

$$\begin{pmatrix} V_x \\ V_y \\ w \end{pmatrix} = BG \begin{pmatrix} V_1 \\ V_2 \\ V_3 \end{pmatrix} \quad (7)$$

x and y estimate is calculated applying an first order approximation, as shown in equations (8) and (9),

$$x(K) = x(K-1) + V_x T \quad (8)$$

$$y(K) = y(K-1) + V_y T \quad (9)$$

where T is the Sampling Time.

2.2 Odometry Error Study

With the objective of evaluate the odometry error a robot race was made, as shown in the flowchart presented in Figure 2. It is possible to observe that the robot moves across several locations, executing the trajectory presented in Figures 3, 4, 5 and 6.

The goal of the controller is to move the robot to a target position with controlled velocity. As input parameters we have as goal the robot displacement to a target position. Initially a position vector pointing to the target position is calculated, the position vector is normalized converting it into a velocity vector, becoming this the objective to accomplish. The equations (1) are used to calculate the velocity that each wheel must have in order to accomplish the objective. At each sampling time the estimated position changes, consequently the position vector changes, the velocity vector changes and the reference speed of each motor changes. The controller has also as objective to follow the trajectory with an angle near to zero. One important fact that needs to be enhanced from the graphics presented in Figures 3, 4 and 5, is that when it is expected the robot to pass by the position $x = 20$ cm and $y = -20$ cm, the robot starts to move to the next target position. This happens because the objective of reaching one position is accomplished if the error in x and in y is less than 2 cm, making the state machine evolve to the next state, changing the objective to $x = 20$ cm and $y = 20$ cm.

The presented graphics (Figures 2, 4, 5, 6 and 7), allow to obtain the odometry error model, relating the acceleration with the odometry error. If the robot accelerates there is an error increase because the acceleration forces the wheels to slip. This situation is

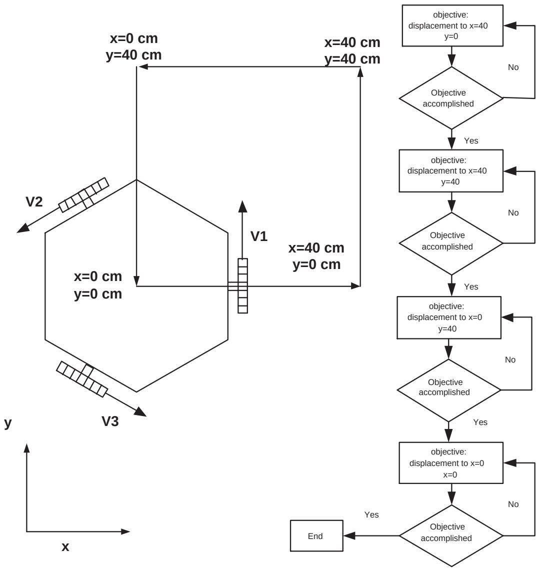

Figure 2: Flowchart of the robot race.

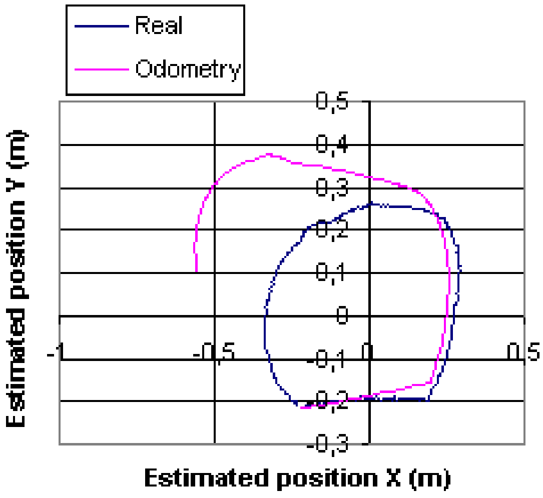

Figure 3: Estimated and real robot trajectory.

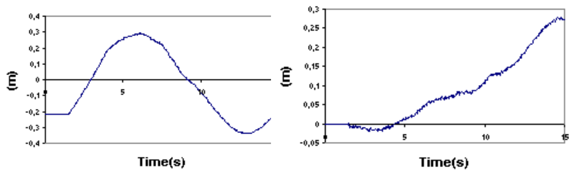

Figure 4: a) Real position x, and (b) Odometry x error.

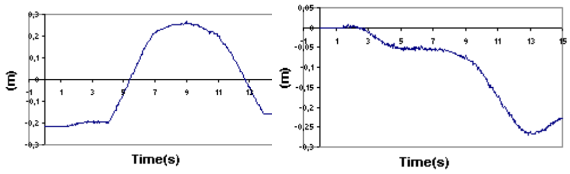

Figure 5: a) Real position y, and (b) Odometry y error.

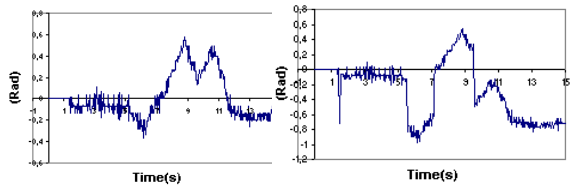

Figure 6: a) Real angle, and (b) Odometry angle error.

more observable when the robot trajectory changes its direction, causing the robot to decelerate and to accelerate causing disturbances in the robot angle, increasing the angle error and consequently changing the rate that the estimate position error evolves.

By this way, if the robot is not moving the variance for x and y is considered null and if the robot is moving the used odometry variance error model for x and y is:

$$Var_{xy} = K_1 + K_2 la(k-1)^2 \tag{10}$$

- K_1 is the variance when the robot is moving in steady state.
- K_2 is a constant that relates the variance with the previous sample time acceleration $la(k-1)$.

The previous sample time acceleration is used instead of the present sample time because it is more representative to evaluate the odometry error noise, because the encoder transitions (necessary to calculate each well velocity estimation resorting to an first order approximation) are updated from the previous sample time up to the present sample time. The angle variance is modeled in a similar way.

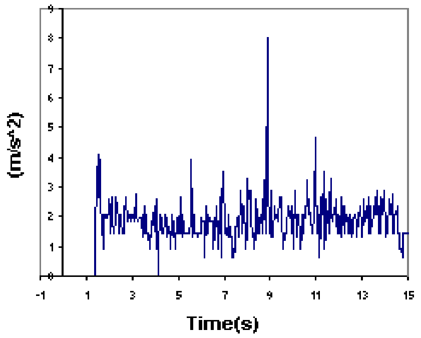

Figure 7: Linear acceleration modulus.

3 ABSOLUTE POSITION ESTIMATION

The Global vision system is required to detect and track a mobile robot in an area supervised by one camera. The camera is placed perpendicular to the ground, fixed to an metallic structure, allowing a maximal height of 3 meters, although in the presented case is placed only at 2 meters height. Placing the camera higher reduces the parallax error, reduces problems such as ball occlusion and the vision field increases, although the image quality decreases and the error due to the barrel distortion effect increases. The image quality concept, in this case study, is related with the number of pixels that are available at each frame to identify and localize an robot marker. The markers, placed on the robot top, have the goal to provide information about the robot localization. Their geometric shape is a circle, all with the same dimensions and with different colors. The number of observed pixels for each marker depends on the illumination conditions, color calibration and camera height. If the camera is placed higher the vision field is bigger, consequently the maximum number of observed pixels for each marker will be reduced.

3.1 Robot Localization

Knowing at first hand that are necessary to localize the robot two different markers, one to identify the center and another to provide information for the angle calculation. Being the field green, and the ball orange, the colors for the robot markers should be the most distant as possible in the RGB cube. The chosen colors were blue for the robot center and yellow for the angle, being the official Robocup colors to distinguish two teams in the SSL placing a colored marker in each robot center (rob, 2008). The ball localization is achieved the same way as the robot center, the only difference is that is a marker placed at a different height and associated to a different color. The blue and yellow markers are used for the robot detection and localization. The blue marker allows x and y calculation, and the yellow marker allows the angle calculation. The used robot is illustrated in Figure 8.

3.2 Global Vision Error Study

It was made for the global vision localization system an analysis of the error probability distributions (Thrun et al., 2005)(Choset et al., 2005). The position error probability distributions were approximated to Gaussian distributions (Ribeiro, 2004)(Negenborn, 2003), being the results presented in SI units. The number of obtained pixels for the blue marker ($Q1$), affects the error variance in x and y, as shown in the next table:

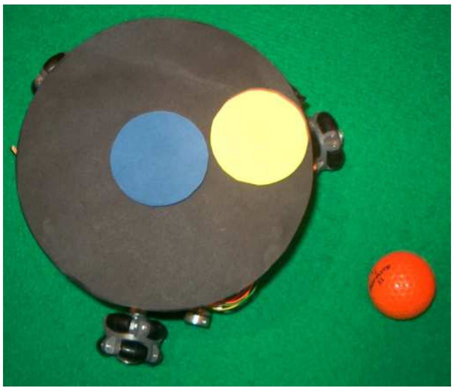

Figure 8: Omnidirectional robot prototype.

Table 1.

$Q1$	x	y
5-10	1,5E-05	1,9E-05
10-20	9,25E-06	7,36E-06
20-30	4,84E-06	4,86E-06
30-40	4,15E-06	3,80E-06
≥ 40	1,96E-06	2,21E-06

On the other hand the variance of the angle error probability distribution is affected by the number of pixels obtained for both makers, for the blue ($Q1$) and for the yellow ($Q2$), as shown in the next tables:

Table 2.

$Q1$	5-10	10-20	20-30	30-40
5-10	0,14	8E-02	1,2E-02	1E-02
10-20	1,6E-02	9,9E-03	1,3E-02	6,6E-03
20-30	1,5E-02	9,9E-03	7,2E-03	4,9E-03
30-40	1,4E-02	9,5E-03	5,9E-03	4,4E-03
≥ 40	1,4E-02	7,2E-03	5,77E-03	3E-03
$Q2$				

Table 3.

$Q1$	≥ 40
5-10	6,2E-03
10-20	4,6E-03
20-30	3,9E-03
30-40	2,9E-03
≥ 40	3E-03
$Q2$	

4 ODOMETRY AND GLOBAL VISION DATA FUSION

Odometry and global vision data fusion was achieved applying an extended Kalman filter. This method was chosen because the robot motion equations are nonlinear and also because the measurements error probability distributions can be approximated to Gaussian distributions (Choset et al., 2005).

4.1 Extended Kalman Filter Algorithm

With the dynamic model given by equations system (7) and considering that control signals change only at sampling instants, the state equation is:

$$\frac{dX(t)}{dt} = f(X(t), u(t_k), t), t\varepsilon[t_k, t_{k+1}] \qquad (11)$$

Where $u(t) = [V_1 V_2 V_3]^T$, that is, the odometry measurements are used as kinematic model inputs. This state should be linearized over $t = t_k$, $X(t) = X(t_k)$ and $u(t) = u(t_k)$, resulting in:

$$A^*k = \begin{pmatrix} 0 & 0 & \frac{-\sin(\theta)}{2\sin(\frac{\pi}{3})} + \frac{\cos(\theta)}{2(1+\cos(\frac{\pi}{3}))} \\ 0 & 0 & \frac{\cos(\theta)}{2\sin(\frac{\pi}{3})} + \frac{\sin(\theta)}{2(1+\cos(\frac{\pi}{3}))} \\ 0 & 0 & 0 \end{pmatrix} \qquad (12)$$

with state transition matrix:

$$\phi^*(k) = exp(A^*(k)(t_k - t_{k-1})) \qquad (13)$$

Resulting in:

$$\phi^*k = \begin{pmatrix} 1 & 0 & (\frac{-\sin(\theta)}{2\sin(\frac{\pi}{3})} + \frac{\cos(\theta)}{2(1+\cos(\frac{\pi}{3}))})T \\ 0 & 1 & (\frac{\cos(\theta)}{2\sin(\frac{\pi}{3})} + \frac{\sin(\theta)}{2(1+\cos(\frac{\pi}{3}))})T \\ 0 & 0 & 1 \end{pmatrix} \qquad (14)$$

Where T is the sampling time $(t_k - t_{k-1})$.

Thus the observations are obtained directly, H^* is the identity matrix.

The extended Kalman filter algorithm steps are as follows (Welch and Bishop, 2001):

1. State estimation at time $t = t_k, X(k^-)$, knowing the previous estimate at $t = t_{k-1}$, $X(k-1)$ and control $u(t_k)$, calculated by numerical integration as shown in equations (5), (8) and (9).
2. Propagation of the state covariance

$$P(k^-) = \phi^*(k)P(k-1)\phi^*(k)^T + Q(k) \qquad (15)$$

Where $Q(k)$ is the noise covariance (11) and also relates to the model accuracy. In order to achieve a more realistic model of the odometry error probability distribution it is necessary to have in account that for abrupt acceleration or deceleration the wheels can slip, consequently there is an significant position estimate error increase, mainly in the angle (Gonçalves et al., 2005).

As there is a measure, the follow also apply:

3. Kalman gain calculation

$$K(k) = P(k^-)H^*(k)^T (H^*(k)P(k^-)H^*(k)^T + R(k))^{-1} \qquad (16)$$

Where $R(k)$ is the covariance matrix of the measurements.

4. State covariation update

$$P(k) = (I - K(k)H^*(k))P(k^-) \qquad (17)$$

5. State update

$$X(k) = X(k^-) + K(k)(z(k) - h(X(k^-, 0))) \qquad (18)$$

Where z(k) is the measurement vector and $h(X(k^-, 0))$ is $X(k^-)$.

4.2 Kalman Filter Performance

With the objective of evaluating the Kalman filter performance another robot race was made, as shown in the flowchart presented in Figure 2. The robot trajectory is presented in Figures 9 and 10.

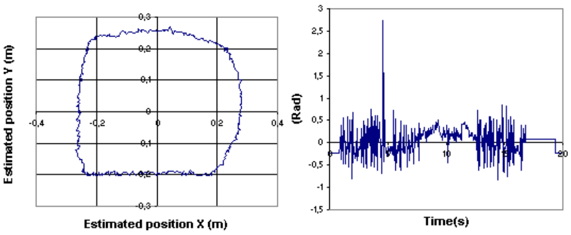

Figure 9: a) Robot trajectory, and (b) Estimated Angle.

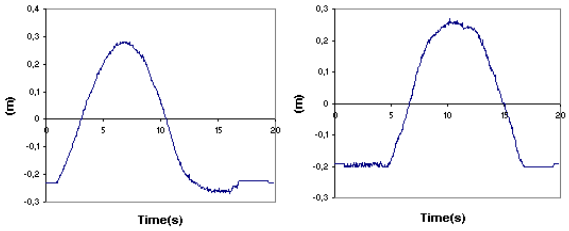

Figure 10: a)Estimated x, and (b) Estimated y.

The image quality of the robot markers for the presented robot race are presented in Figure 11 and the variance of the estimated robot position is presented in Figure 12. Whenever the image quality decreases the variance error of position estimate increases compromising the controller performance. On the other hand whenever the image quality increases the error variance is reduced and when the state update is done the position estimate error is reduced.

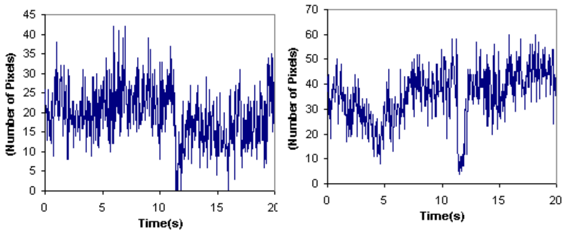

Figure 11: a) Image quality of the center marker Q1, b)Image quality of the angle marker Q2.

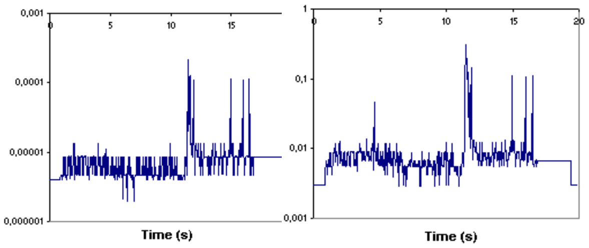

Figure 12: a) x and y variance, and (b) Angle variance.

5 CONCLUSIONS

Omnidirectional vehicles have many advantages in robotics soccer applications, allowing movements in every direction. The fact that the robot is able to move from one place to another with independent linear and angular velocities contributes to minimize the time to react, the number of maneuvers is reduced and consequently the game strategy can be simplified.

The robot relative position estimation is based on the odometry calculation. The odometry calculation uses each wheel velocity in order to estimate the robot position, the disadvantage is that the position estimate error is cumulative and increases over time.

It was made for the global vision localization system an analysis of the error probability distributions. The number of obtained pixels for the blue marker ($Q1$), affects the error variance in x and y. On the other hand the variance of the angle error probability distribution is affected by the number of pixels obtained for both makers, for the blue ($Q1$) and for the yellow ($Q2$).

Odometry and global vision real time data fusion was achieved applying an extended Kalman filter. This method was chosen because the robot motion equations are nonlinear and also because the measurements error probability distributions can be approximated to Gaussian distributions.

REFERENCES

(2008). Robocup. http://www.robocup.org/.

Borestein, Everett, and Feng (1996). *where am I, Sensores and Methods for Mobile Robot Positioning*. Prepared by the University of Michigan.

Choset, H., Lynch, K., Hutchinson, S., Kantor, G., Burgard, W., Kavraki, L., and Thrun, S. (2005). *Principles of Robot Motion : Theory, Algorithms, and Implementations*. MIT Press.

Dudek, G. and Jenkin, M. (2000). *Computational Principles of Mobile Robotics*. Cambridge University Press.

Gonçalves, J., Costa, P., and Moreira, A. (2005). *Controlo e estimação do posicionamento absoluto de um robot omnidireccional de três rodas*. Revista Robótica, Nr 60, pp 18-24.

Gonçalves, J., Pinheiro, P., Lima, J., and Costa, P. (2007). Tutorial introdutório para as competições de futebol robótico. *IEEE RITA - Latin American Learning Technologies Journal*, 2(2):63–72.

Kalmár-Nagy, T., D'Andrea, R., and Ganguly, P. (2002). Near-optimal dynamic trajectory generation and control of an omnidirectional vehicle. In *Sibley School of Mechanical and Aerospace Engineering*.

Negenborn, R. (2003). *Robot Localization and Kalman Filters - On finding your position in a noisy world*. Master Thesis, Utrecht University.

Ribeiro, F., Moutinho, I., Silva, P., Fraga, C., and Pereira, N. (2004). Controlling omni-directional wheels of a robocup msl autonomous mobile robot. In *Proceedings of the Scientific Meeting of the Robotics Portuguese Open*.

Ribeiro, M. I. (2004). *Gaussian Probability Density Functions: Properties and Error Characterization*. Technical Report, IST.

Sousa, A. (2003). *Arquitecturas de Sistemas Robóticos e Localização em Tempo Real Através de Visão*. PHD Thesis, Faculty of Engineering of the University of Porto.

Thrun, S., Burgard, W., and Fox, D. (2005). *Probabilistic robotics*. MIT Press.

Welch, G. and Bishop, G. (2001). *An introduction to the Kalman filter*. Technical Report, University of North Carolina at Chapel Hill.