Proceedings of the 8th International
IEEE Conference on Intelligent Transportation Systems
Vienna, Austria, September 13-16, 2005



# Sensor Fusion for Precise Autonomous Vehicle Navigation in Outdoor Semi-structured Environments

L. Conde Bento, Urbano Nunes, Fernando Moita and António Surrecio

*Abstract*— **This paper presents a guidance system for autonomous vehicles navigation in semi-structured outdoor environments. It integrates redundant encoders data and absolute positioning data provided by landmarks and artificial beacons. Natural features are localized using a laser range sensor, and magnetic sensing rulers were developed to detect magnetic markers buried in the ground. In the first fusion stage, data from four wheel encoders and one steering encoder are fused by means of an EKF, providing robust odometric information, namely in face of undesirable effects of wheels slippage. Next, a second fusion stage is processed for integrating odometric and absolute positioning data. Simulation and real experiments using a four-wheels actuated electrical vehicle are presented.**

## I. INTRODUCTION

Odometry being essential for autonomous navigation, is not enough due to its relative and integrative nature. So, it is required to complement odometric data with absolute positioning. Due to systematic error problems, some effort has been dedicated in modelling the uncertainty propagation in order to get somehow a reliable odometry model [6]. Data fusion of ABS sensors and GPS for outdoor localization, based on an Extended Kalman Filter (EKF) had been presented by Bonnifait, et al. [2].

For absolute positioning using landmarks many laser and vision based systems have been developed. Self localization given a map of the environment and the more challenging problem of simultaneous localization and mapping (SLAM) are two examples of key mobile robot problems requiring absolute/relative positioning data. The most commonly used localization probabilistic approaches employ Kalman filtering (e.g [8]), grid-based Markov localization [5], and Monte Carlo methods [4]. Gutmann and Fox tested six different localisation methods in a Robocup-like environment as reported in [7]. On the other hand the California Partners for Advanced Transit and Highways (PATH) Program has been given important contributes in the development of a reference system based on magnets for vehicle lateral guidance/control [12].

The paper addresses research on the development of a precise autonomous navigation system applied on guiding a four-wheel actuated electrical vehicle moving in semi-structured outdoor environments. Its purpose is to provide guidance control with anti-collision behaviour for low speed vehicles moving in cybercars scenarios. Cybercars [3] have to satisfy challenging requirements like following a path, with high accuracy, in narrow spaces shared with other vehicles and in some areas with pedestrians, providing ride comfort, with low-level of jerk, and assuring complete safety with human driverless control. The overall navigation system (see Fig. 1) is composed by three main subsystems, which are designated by path-following controller (PFC), vehicle's pose estimator (VPE) and multi-target detection and tracking system (MTDTS). The PFC and MTDTS systems are described in [1] and [9], respectively. This paper addresses the VPE module.

## II. ODOMETRY MODEL AND FUSION

We are using a Robucar (manufactured by Robosoft) in our autonomous navigation experiments. It is equipped with four wheels, each one driven by an independent motor equipped with its own encoder. It is a bi-steerable car, but in our models and experiments, only front steering has been used.

Let the vehicle position be represented by the middle point rear axle $M$ with Cartesian coordinates $(x_k, y_k)$ at time $t_k$, as shown in Fig. 2. The vehicle local coordinates system is defined as having origin $M$ and its $x$-axis aligned with the longitudinal axis of the car. $\theta_k$ is the vehicle heading angle at time $t_k$. Assuming that the vehicle's motion is locally circular, its position and orientation at time $t_k$ is given by

$$\begin{cases} x_{k+1} = x_k + \Delta\cos(\theta_k + \omega/2) \\ y_{k+1} = y_k + \Delta\sin(\theta_k + \omega/2) \\ \theta_{k+1} = \theta_k + \omega \end{cases} \quad (1)$$

where $\Delta$ is the arc length and $\omega$ the elementary rotation. Assuming that there is no wheels slippage and using only data from the rear wheels encoders, then

$$\Delta = \frac{\Delta_{RR}+\Delta_{RL}}{2} \quad , \; \omega = \frac{\Delta_{RR}-\Delta_{RL}}{2e} \quad (2)$$

where $e$ is the half distance between wheels, and $\Delta_{RR}$ and $\Delta_{RL}$ are calculated using the right and left wheel encoders measurements, respectively. Using also the front wheels encoders, redundant data become available, which can be used to produce better estimates of $\Delta$ and $\omega$. The steering angle of left and right wheels can be expressed by

$$\Psi_L = \arctan\left(\frac{\tan(\Psi).L}{L - e.\tan(\Psi)}\right) \quad (3)$$

$$\Psi_R = \arctan\left(\frac{\tan(\Psi).L}{L + e.\tan(\Psi)}\right) \quad (4)$$

This work was supported by Institute of Systems and Robotics and Fundação para a Ciência e Tecnologia under grant POSC/EEA-SRI/58016/2004

L. Conde Bento is with Institute of Systems and Robotics, and Polytechnic Institute of Leiria conde@isr.uc.pt

Urbano Nunes is with Institute of Systems and Robotics, and University of Coimbra urbano@isr.uc.pt

Fernando Moita is with Institute of Systems and Robotics, and Engineering Institute of Coimbra moita@isr.uc.pt

António Surrecio is with Institute of Systems and Robotics surrecio@isr.uc.pt

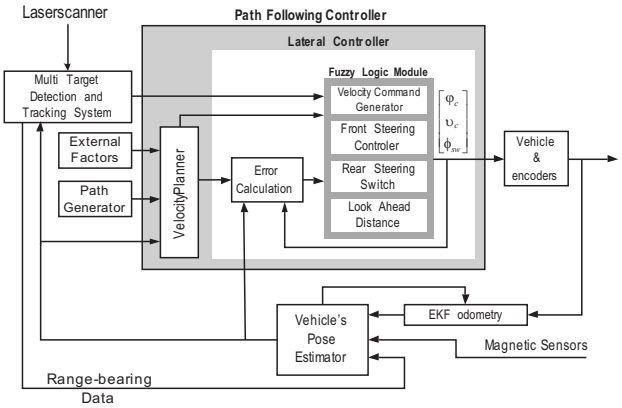



Fig. 1. Navigation system architecture. The dashed part is only used in the simulation process.

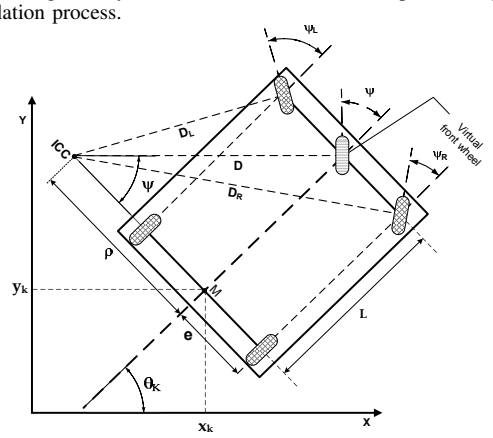



Fig. 2. Vehicle geometrical configuration.

where $\Psi$ is the steering angle of the virtual front wheel and $L$ is the car length (distance between rear and front axles). The vehicle geometrical configuration parameters are illustrated in Fig. 2, and are summarized in Table I. From (3) and (4) and knowing that $\Delta = \rho.\omega$, a set of equations can be established which relates the encoders measurements (from each of the four wheels and steering) with the parameters $\Delta$ and $\omega$ [2]:

$$\begin{cases} \tan(\Psi) = L.\frac{\omega}{\Delta} \\ \Delta_{RL} = \Delta - e.\omega \\ \Delta_{RR} = \Delta + e.\omega \\ \Delta_{FL}.\cos(\Psi_L) = \Delta - e.\omega \\ \Delta_{FR}.\cos(\Psi_R) = \Delta + e.\omega \end{cases} \tag{5}$$

Therefore, considering $\mathbf{x} = [\Delta, \omega]^T$ and $\mathbf{y} = [\tan(\Psi), \Delta_{RL}, \Delta_{RR}, \Delta_{FL}.\cos(\Psi_L), \Delta_{FR}.\cos(\Psi_R)]^T$ the state and measurement vectors, respectively, an EKF can be applied as in [2] to estimate the state vector, from the redundant data.

## III. ODOMETRY CONFIDENCE TESTS AND SIMULATION

In this section we apply the EKF odometry computation described in the previous section. Real data measurements, gathered from Robucar encoders moving along a closed path as depicted in Fig. 3, were used in the reported simulations.

TABLE I
VEHICLE GEOMETRICAL CONFIGURATION PARAMETERS

| | |
|---|---|
| $L$ | car length (distance between rear and front axle) |
| $e$ | half-track (half car width) |
| $\Psi_R$ | steering angle from right wheel |
| $\Psi_L$ | steering angle from left wheel |
| $\Psi$ | steering angle of the virtual front wheel |
| $D$ | curvature radius of virtual front wheel |
| $\rho$ | curvature radius of the rear axle center |

The qualitative behavior of the EKF odometric model is very satisfactory in normal road conditions of adherence as illustrated in Fig. 3-(a) and 3-(b). We added white Gaussian noise to the measures with a signal-to-noise ratio of 10dB. The EKF filters efficiently the added noise. This result is well illustrated in Fig. 3-(c), where it is shown that the EKF odometry approaches very closely the real trajectory (computed with the encoders data without noise), while the trajectory computed from the noisy measurements diverges a lot, as expected.

However this odometric model doesn't solve the problem inherent to slippages. If a big slippage occurs, the Kalman filter will not eliminate its effects. This problem can be attenuated by pre-processing the redundant data before providing it to the EKF odometry algorithm. We can compute an approximate motion of the rear wheels based on the motion performed by the front wheels and vice versa, applying the following equations:

$$\begin{array}{l} \Delta_R = \frac{\Delta_{RR}+\Delta_{RL}}{2} \ ; \ \omega_R = \frac{\Delta_{RR}-\Delta_{RL}}{2e} \\ \Delta_F = \frac{\Delta_{FR}+\Delta_{FL}}{2} \ ; \ \omega_F = \frac{\Delta_{FR}-\Delta_{FL}}{2e} \end{array} \tag{6}$$

$$\Delta_{F/R} = \frac{\Delta_R}{\cos(\Psi)} \ ; \ \Delta_{R/F} = \cos(\Psi).\Delta_F \tag{7}$$

$$\begin{array}{l} \Delta_{VRL} = \Delta_{R/F} - e.\omega_F \ ; \ \Delta_{VRR} = \Delta_{R/F} + e.\omega_F \\ \Delta_{VFL} = \Delta_{F/R} - e.\omega_R \ ; \ \Delta_{VFR} = \Delta_{F/R} + e.\omega_R \end{array} \tag{8}$$

In (6) ($\Delta_R$,$\omega_R$) and ($\Delta_F$,$\omega_F$) are the parameters in respect to the midpoint of the rear axle and front axle, respectively. In (7) and (8) $\Delta_{i/j}$ means $\Delta_i$ computed based on measurements from $j$, with $i, j = \{F, R\}$. Equations (8) express the designated virtual displacements for each wheel. Based on (8) we define the following confidence coefficients:

$$\begin{array}{l} CC_R = 1 - \frac{|\Delta_{VRL}-\Delta_{RL}|+|\Delta_{VRR}-\Delta_{RR}|}{|\Delta_{VRL}+\Delta_{RL}+\Delta_{VRR}+\Delta_{RR}|} \\ CC_F = 1 - \frac{|\Delta_{VFL}-\Delta_{FL}|+|\Delta_{VFR}-\Delta_{FR}|}{|\Delta_{VFL}+\Delta_{FL}+\Delta_{VFR}+\Delta_{FR}|} \end{array} \tag{9}$$

The confidence coefficients are used to decide if a virtual measure is used instead of the real measure. Fig. 3-(d,e,f) show results of using the algorithm in a simulation of slippages injected on the rear right wheel at $t = 10s$ and $t = 50s$. As we can see from Fig. 3-d, the EKF with this data pre-processing, henceforth known as EKF-CT odometry (EKF odometry with confidence tests), will not follow the slippage, hence the algorithm detected a wrong measure and replaced it by the virtual measure, computed based on the

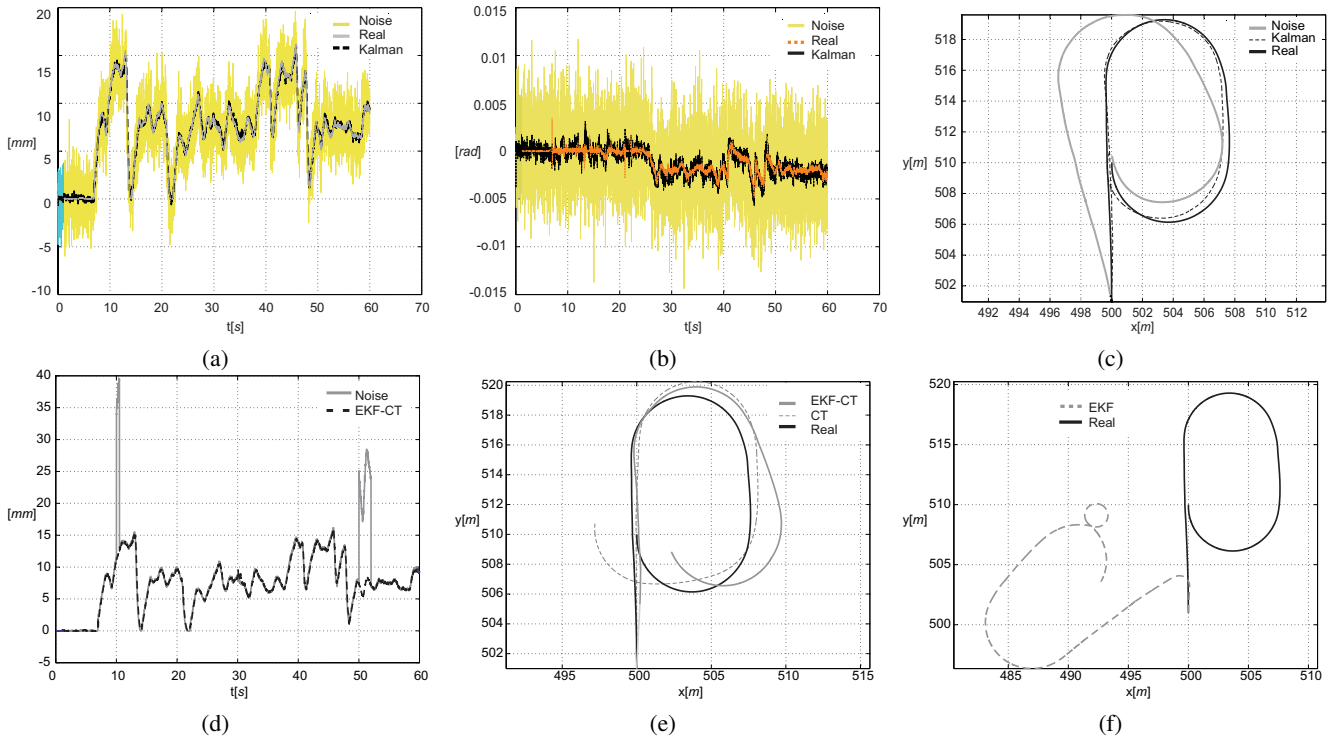



Fig. 3. (a) EKF estimation for $\Delta$ (Real - encoders measures; Noise - measures with white noise; Kalman - obtained estimation) (b) EKF estimation for $\omega$ (Real - encoders measures; Noise -measures with white noise; Kalman - obtained estimation) (c) Odometry results with and without the EKF (Real - encoders measures; Noise - measures with white noise; Kalman - obtained estimation) (d) $\Delta$ estimation with a simulated slippage (Noise - encoders measures with a simulated slippage at t=10 s and t= 50 s; EKF-CT - estimation with EKF-CT) (e) Odometry obtained with CT (CT - path estimated applying the CT and odometry model (1); Real - path without the simulated slippage; EKF-CT - path estimated applying the EKF-CT) (f) Odometry obtained with EKF without using CT (Real - path without the simulated slippage; EKF - path estimated applying the EKF odometry).

other measures. Estimated paths using the EKF-CT and the EKF algorithms are illustrated in Fig. 3-e and 3-f. In both, the continuous line represents the real path. In Fig. 3-e the dotted line represents the estimated path using odometry model (1) and confidence tests.

## IV. FUSION OF ODOMETRY AND ABSOLUTE POSITIONING DATA

The vehicle is equipped with sensors which provide absolute positioning data: 1) a SICK laser which provides range-bearing data $(d, \phi)$ associated to visible landmarks; 2) two magnetic sensing rulers, one placed on the front and the other on the rear of the vehicle. The magnetic sensing rulers developed at ISR [10], based on an array of adjacent Hall effect sensors, detect robustly magnetic markers which are placed in the ground defining centerpoints of the path to be followed by the vehicle.

The fusion of odometry with absolute positioning data is made by means of EKFs. The vehicle's pose defined by the Cartesian coordinates $(x, y)$ and heading $(\theta)$ are the state variables of the EKF. The state variables of the EKF odometry (section II) are here treated as inputs to the EKF

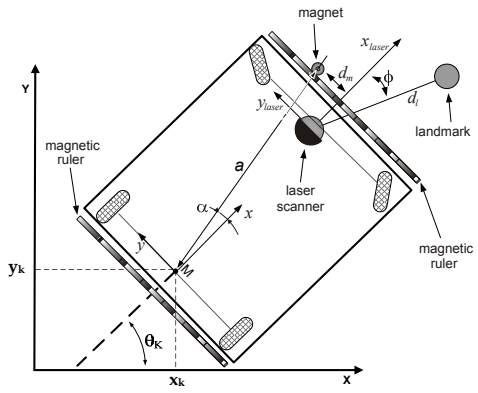



Fig. 4. Measurement model variables. In figure, $(a, \alpha)$ denotes range-bearing data associated to a magnet detection.

data fusion, i.e $\mathbf{u}_k = (\Delta, \omega)$ with associated noise covariance matrix $\mathbf{\Gamma}_k$. The range-bearing measurements associated to each landmark are treated as measurements in the fusion process.

**1)** ***System Model:*** the system model is defined by the kinematic nonlinear equations (1), with state vector $\mathbf{x}_k =$

$[x_k\ y_k\ \theta_k]^T$, and input $\mathbf{u}_k = [\Delta_k\ \omega_k]^T$, which can be written in the compact form (including noises):

$$\mathbf{x}_k = \mathbf{f}(\mathbf{x}_{k-1}, \mathbf{u}_k, \gamma_k, \sigma_k) \tag{10}$$

where $\gamma_k$ and $\sigma_k$ denote the system and input noises, with associated matrices $\mathbf{Q}$ and $\mathbf{\Gamma}_k$.

**2)** ***Measurement Model*** *(example for the front magnetic ruler and laser-based detected landmarks):* let $(a, \alpha)$ be the range-bearing pair, associated to a detected landmark, defined in the local robot coordinate system (see Fig.4). Thus the following equations yield:

$$\begin{array}{c} a = \sqrt{(y_f - y_k)^2 + (x_f - x_k)^2} \\ \alpha = \arctan \frac{y_f - y_k}{x_f - x_k} - \theta_k \end{array} \tag{11}$$

where $(x_f, y_f)$ represents the Cartesian position of the landmark. From (11) we can define the nonlinear measurement model

$$\mathbf{z}_k = \mathbf{h}(\mathbf{x}_k) + \mathbf{v}_k \tag{12}$$

where $\mathbf{h}(\mathbf{x}_k)$ is the nonlinear vector function

$$\mathbf{h}(\mathbf{x_k}) = \left[ \begin{array}{c} \sqrt{(y_f - y_k)^2 + (x_f - x_k)^2} \\ \arctan \frac{y_f - y_k}{x_f - x_k} - \theta_k \end{array} \right] \tag{13}$$

and $\mathbf{v}_k$ is the Gaussian sensor noise vector with covariance matrix $\mathbf{R_k}$. The range-bearing data $(a\ \alpha)$ are the observation values entering the EKF, $\mathbf{z} = [a\ \alpha]^T$, which are calculated from sensor measures as follows:

**a)** *for magnetic marker*

$$\begin{array}{l} a = \sqrt{d_m^2 + L_1^2} \\ \alpha = \arctan \frac{d_m}{L_1} \end{array} \tag{14}$$

where $d_m$ is the magnetic ruler measure which corresponds to the distance between the marker with known position $(x_m, y_m)$ and the magnetic-sensing ruler central point, and $L_1$ is the distance between the front magnetic ruler and the vehicle rear axis (we are assuming that the ruler is perfectly parallel with the $y$-axis of robot coordinate system).

**b)** *for laser-based detected landmark*

$$\begin{array}{l} a = \sqrt{d_l^2 + L_l^2 + 2 d_l L_l \cos(\phi)} \\ \alpha = \arctan \frac{d_l \sin(\phi)}{L_l + d_l cos(\phi)} \end{array} \tag{15}$$

where $(d_l, \phi)$ are the range-bearing data described in the laser coordinate system. It is assumed that the laser coordinate system is aligned with the robot coordinate system, with a distance $L_l$, defined in the $xy$-plane, between them.

Another (non-standard) measurement model has been investigated and applied as described in [11], which consists on considering in model (12):

$$\begin{array}{l} \mathbf{z} = [x_f\ y_f]^T \\ \mathbf{h}(\mathbf{x}_k) = \left[ \begin{array}{c} x_k + a\cos(\theta_k + \alpha) \\ y_k + a\sin(\theta_k + \alpha) \end{array} \right] \end{array} \tag{16}$$

**3)** ***EKF algorithm*****:** it is composed by the following prediction and correction stages:

***Prediction stage***

$$\begin{array}{c} \hat{\mathbf{x}}_k^- = f(\hat{\mathbf{x}}_{k-1}, \mathbf{u}_k, 0, 0) \\ \mathbf{P}_k^- = \mathbf{A}_k \mathbf{P}_{k-1} \mathbf{A}_k^T + \mathbf{B}_k \mathbf{\Gamma}_{k-1} \mathbf{B}_k^T + \mathbf{Q} \end{array} \tag{17}$$

where the system ($\mathbf{A}$) and input ($\mathbf{B}$) matrices are calculated as the following Jacobian of the system $\mathbf{f}(.)$ function:

$$\begin{array}{l} \mathbf{A}_k = \left[ \begin{array}{ccc} 1 & 0 & -\Delta_k \sin(\theta_k + \frac{\omega_k}{2}) \\ 0 & 1 & \Delta_k \cos(\theta_k + \frac{\omega_k}{2}) \\ 0 & 0 & 1 \end{array} \right] \\ \mathbf{B}_k = \left[ \begin{array}{cc} \cos(\theta_k + \frac{\omega_k}{2}) & -\frac{\Delta_k}{2} \sin(\theta_k + \frac{\omega_k}{2}) \\ \sin(\theta_k + \frac{\omega_k}{2}) & \frac{\Delta_k}{2} \cos(\theta_k + \frac{\omega_k}{2}) \\ 0 & 1 \end{array} \right] \end{array} \tag{18}$$

***Correction stage***

Once measurements $(a, \alpha)$ become available the following correction stage is done:

$$\begin{array}{l} \mathbf{S}_k = (\mathbf{H}_k \mathbf{P}_k^- \mathbf{H}_k^T + \mathbf{R}_k) \\ \mathbf{K}_k = \mathbf{P}_k^- \mathbf{H}_k^T \mathbf{S}_k^{-1} \\ \hat{\mathbf{x}}_k = \hat{\mathbf{x}}_k^- + \mathbf{K}_k(\mathbf{z}_k - \mathbf{h}(\hat{\mathbf{x}}_k^-)) \\ \mathbf{P}_k = (\mathbf{I} - \mathbf{K}_k \mathbf{H}_k) \mathbf{P}_k^- \end{array} \tag{19}$$

where $\mathbf{I}$ is a identity matrix and $\mathbf{H}_k$ is the Jacobian of the measurement $\mathbf{h}(.)$ function:

$$\mathbf{H}_k = \nabla_x \mathbf{h}(\mathbf{x}_k) \tag{20}$$

### *A. Data Association*

In this work we have adopted the conventional *nearest neighbour* data association method, using the normalized innovation distance

$$\bar{d}_k = {\upsilon_k}^T {\mathbf{S}_k}^{-1} \upsilon_k \tag{21}$$

where $\mathbf{v}_k$ denotes the innovation sequence $\upsilon_k = \mathbf{z}_k - \mathbf{h}(\hat{\mathbf{x}}_k^-)$ and $\mathbf{S}_k$ its predicted covariance, for accepting/rejecting observations.

## V. SIMULATION RESULTS

### *A. Magnet Based*

The VPE based on fusion of odometry and landmarks, described in previous sections was extensively simulated. Some results are shown and discussed in this section. In the reported simulations two types of disturbances are considered: Systematic errors and Gaussian sensors noise.

Systematic errors were applied in the process by multiplying $\Delta$ by a factor $K$, yielding

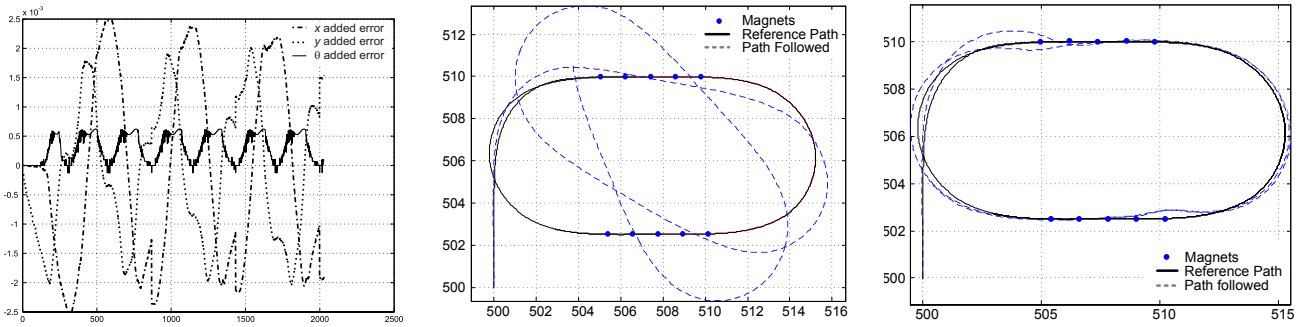
$$\Delta_{WN} = \Delta \times K \tag{22}$$

By this way, uncertainty is introduced in the pose $(x_k, y_k, \theta_k)$ obtained from the odometry computation. In order to evaluate the magnitude of the errors introduced by systematics errors, the magnitude of the disturbance in the vehicle's pose is displayed in Fig. 5 (a). The noise added to each component of the vehicle's pose was computed by the following way:

$$[x_e, y_e, \theta_e] = f(\Delta_{WN}, \omega) - f(\Delta, \omega) \tag{23}$$

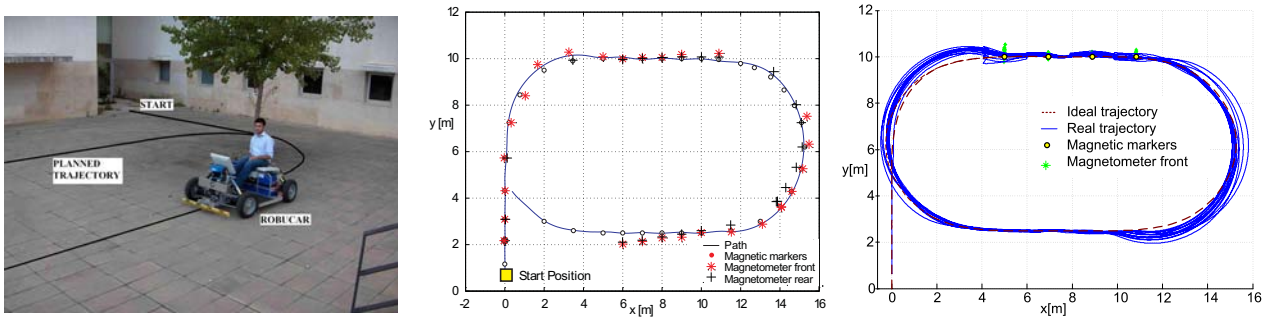
Additionally, Gaussian noise was added, denoted by $C$, resulting

$$\left\{ \begin{array}{l} x_{k+1} = x_k + \Delta \times K \times \cos(\theta_k + \omega/2) + C \\ y_{k+1} = y_k + \Delta \times K \times \sin(\theta_k + \omega/2) + C \\ \theta_{k+1} = \theta_k + \omega + C \end{array} \right. \tag{24}$$




(a) (b) (c)

Fig. 5. (a) Systematic noise added to the pose ($\Delta_{WN}$ calculated using (22) with $K = 1.03$) in meters; (b) Odometry results without EKF corrections. (c) VPE using EKF fusion of odometry and ten magnetic markers (five markers on each side of the loop).




(a) (b) (c)

Fig. 6. (a) Field test environment (b) Experiment without odometry calibration – 1 run, Robucar moving autonomously. In this experiment 36 magnets were used. The path followed by the Robucar was recorded using data from the front magnetic ruler (*), rear magnetic ruler (+) and data from the encoders (continuous line), (standard odometry (1)) (c) Experimental results obtained using only the front magnetic ruler – Robucar moving autonomously under a fuzzy path-following controller [1] . The four bullets represent the four physical magnetic markers used in the experiment.

In real environments the detection of the magnets doesn't return the exact center of the magnet, but rather a coordinate close to its $z$-axis center, so in order to have simulated measures similar to real ones, a representative model of the magnetic field radiated by the magnetic marker was used in simulations. Thus a magnetic marker was modeled as a magnetic dipole with the magnetic field, $B(x, y, z)$, at an arbitrary point $P(x, y, z)$, expressed as follows (in cgs units):

$$B = \frac{\mu_0 M}{4\pi r^5}(3xz\hat{i} + 3yz\hat{j} + (2z^2 - x^2 - y^2)\hat{k}) \qquad (25)$$

where $r = \sqrt{x^2 + y^2 + z^2}$, $M$ is the magnetic moment of the magnetic marker, and the $z$-axis corresponds to the height relative to the marker center.

Results shown in Fig. 5-b and 5-c exemplify how the fusion process leads to a correct vehicle path following. If no correction is done, the path actually followed (path followed) is much different from the reference path. In Fig. 5-c the EKF handles with the disturbances by using the magnets located at the marked points. Although the errors are accumulated during the curves, on the straight lines it recovers by using the detected magnets. The fusion method also handles false detections either coming from hardware anomaly or from incorrect positioned magnets.

### B. Laser based

In the laser sensor emulations, a maximum value of 8 meters and an angular field of view of 180 degrees was considered. Fig. 7 shows simulation results of a path following, with the control system affected by a disturbance signal composed by: (1) Gaussian white noise added to the encoder data with variance one; (2) distance dependent Gaussian white noise added to the laser data; and (3) impulsive disturbances to simulate wheels slippage. In order to simulate the data association process, three false landmarks were placed, as depicted in Fig. 7. The false landmarks, unknown in the environment model, are used to simulate false detections or bad associations. When one of these landmarks are detected, the association to the nearest known landmark yields a high innovation absolute value, thus resulting that the false landmark is discarded by the validation gate process as illustrated in Fig. 8. The threshold used was 0.01.

## VI. Experimental Results

Extensive simulations have been done showing the effectiveness of the proposed VPE data fusion module. Field experiments have also been done, with the purpose of analysing the localization system behavior. Whenever a sensor ruler detects a magnetic marker, the measure ($d_m$) enters the data

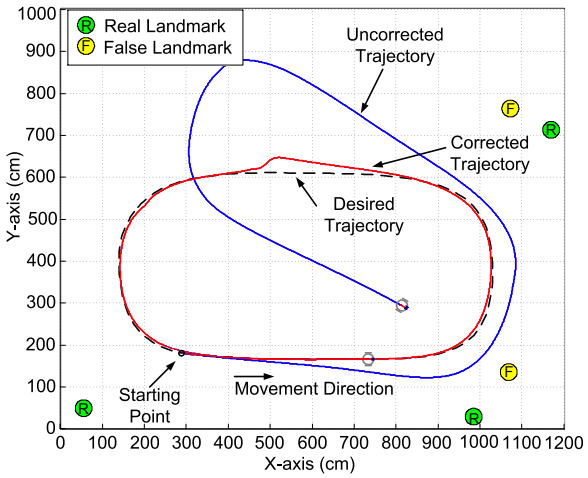



Fig. 7. Simulation runs for laser-based VPE.

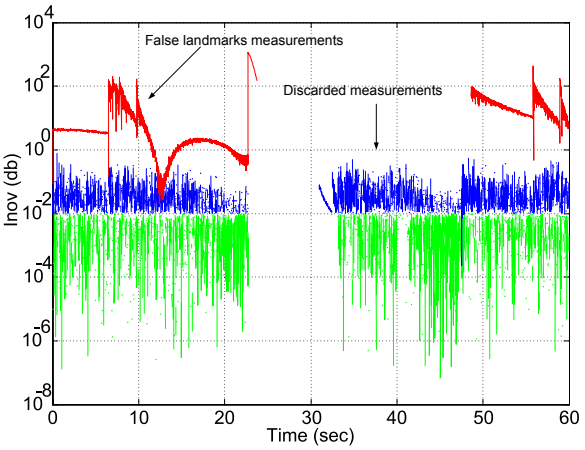



Fig. 8. Validation gate. The values over $10^{-2}$ represent: in red – false landmarks measurements; in blue – discarded real measurements.

fusion algorithm and is accepted or not depending of the validation gate result.

Two types of experiments are reported in this section. Both concern the path-following control of a Robucar moving along a predefined closed path (see Fig. 6-a, 6-b and 6-c. Fig. 6-a shows the test field environment where the virtual line represents a rough approximation of the planned trajectory. In both cases the same fuzzy path-following controller, described in [1], was used. However, in one case, the detected magnetic markers information was used in the on-line computation of the vehicle's pose, and so used in the calculation of the errors to the controller, and in the other case it wasn't used neither in the vehicle's pose estimation nor in the controller. As shown in Fig. 6-c, when the vehicle passes through a magnetic marker, the odometry is calibrated based on the detected lateral deviation of the vehicle. In the experimental result shown in figure, only the front magnetic ruler and only four magnets, all aligned on one side of the loop, were used. This simple configuration was enough to keep the Robucar tracking the path. In a way to compare the performance of the navigation system, with the calibration method based on the detected markers, Fig. 6-b illustrates an experimental test where no calibration was done to the odometry system. In the same figure, the continuous line represents the computed path as obtained by the odometry system; and the stars marks represent the path followed by the center of the front magnetic ruler, while the cross marks represent the path followed by the center of the rear magnetic ruler. As can be observed, the Robucar lost track of the path at the first loop run, when finishing the loop. Notice that the main error is in its orientation, hence the car before losing the track of the path was following a straight line but with a wrong orientation.

## VII. Conclusions And Future Work

The experimental odometry calibration with the magnetic sensing rulers revealed good results. We are now testing extensively the complete fusion process, integrating also position data from detected natural features.

The majority of systematic errors associated to the odometry relying only on encoders, are by the markers based calibration, somehow eliminated. However that procedure by itself does not solve the slippage (or high slippage) problem, which can be reduced by applying confidence tests as proposed in Section III.

More field experiments are being carried out to deeply characterize the performance of the overall VPE data fusion module.

## References


[1] L. Conde Bento and Urbano Nunes. Autonomous navigation control with magnetic markers guidance of a cybernetic car using fuzzy logic. *Machine Intelligence and Robotic Control*, in press.

[2] P. Bonnifait, P. Crubill, and D. Meizel. Data fusion of four abs sensors and gps for an enhanced localization of car-like vehicles. In *IEEE Int. Conf. on Robotics and Automation*, pages 1597–1602, 2001.

[3] Cybercars. Cybernetic technologies for the car in the city. [online], www.cybercars.org, 2001.

[4] D. Fox, W. Burgard, F. Dellaert, and S. Thrun. Monte Carlo Localization: efficient position estimation for mobile robots. In $16^{th}$ *National Conf. on Artificial Intelligence (AAAI99)*, USA, 1999.

[5] D. Fox, W. Burgard, and S. Thrun. Markov localization for mobile robots in dynamic environments. *Journal of Artificial Intelligence Research*, 11, 1999.

[6] J. Hermosillo, C. Pradalier, and S. Sekhavat. Modeling odometry and uncertainty propagation for a bi-steerable car. In *IEEE Proc. of Intelligent Vehicles Symp. (IV2002)*, Versailles, France, 2002.

[7] Jens-Steffen and Dieter Fox. An experimental comparation of localization methods continued. In *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, pages 454–459, Lausanne, Switzerland, 2002.

[8] J. Leonard and H. F. Durrant-Whyte. Mobile robot localization by tracking geometric beacons. *IEEE Trans. on Robotics and Automation*, 7(3):376–382, 1999.

[9] Abel Mendes and Urbano Nunes. Situation-based multi-target detection and tracking with laserscanner in outdoor semi-structured environment. In *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, pages 88–93, Sendai, Japan, 2004.

[10] Fernando Moita and Urbano Nunes. Magnetic ruler version 1.0. configuration, software structure and characterization. Technical report ISRLM2004/03, Institute of Systems and Robotics, Portugal, 2004.

[11] A. Surrecio, U. Nunes, and R. Araujo. Fusion of odometry with magnetic sensors using kalman filters and augmented system models for mobile robot navigation. In *Proc. of the IEEE Int. Conf. on Industrial Electronics*, Croatia, 2005.

[12] W. Zhang and Robert E. Parson. An intelligent roadway reference system for vehicle lateral guidance/control. In *Proc. of the 1990 American Control Conf.*, pages 281–286, USA, 1990.