Control system and laser-based sensor design of an autonomous vehicle for industrial environments

José Miguel Sanchiz, Jorge Badenas and Filiberto Pla

University Jaume I, Av. Sos Baynat, 12071 Castelló, Spain

ABSTRACT

This work presents an approach to the sensorial device and control system of an autonomous vehicle intended for navigating and performing precise load/unload tasks in industrial environments. The control system is able to perform turns, line following, and arbitrary curve following specified as splines. It is based on a multivariable design using the technique of pole placement in state space. The control system uses results from parameter estimation modules to adapt to the changing responses of traction motors when loaded or unloaded, such estimators are Kalman filters that recover the vehicle motion parameters from measurements performed by the positioning sensor. Several steering configurations are possible since the control system provides a radius of turn as output. So differential drive, tricycle drive or Ackerman steering can be done by transforming this radius in motor orders, depending on the geometry of the vehicle. The only sensor the system relies on is a laser-based local positioning system consisting of a rotating laser and retro-reflectors. Robust algorithms for signal analysis and position/orientation estimation have been developed. The sensor is able to detect reflectors 25 meters away in daylight or in dusty industrial environments using a low-cost 1 mW laser. The system has been tested on two mobile bases, using differential drive and tricycle drive.

Keywords: Autonomous vehicles, laser positioning sensors, control systems, path following

1. INTRODUCTION

One of the main applications of autonomous vehicles is to accomplish load/unload operations in industrial environments. In these applications one or several vehicles perform constant product storage or deliver unfinished products to a production line. This is the case of ceramics manufacturing, for example, where these systems are quite introduced. Another possible sector is in agricultural manipulations, like fruits, where products are sorted and stored. In this sector autonomous vehicles are yet not so introduced probably because, unlike in ceramics, the variety of storage devices and storage places, where products can be stored in cooled cameras, stacked, etc, poses the need for highly intelligent and flexible systems.

In this work we describe the sensorial and control parts of an autonomous vehicle intended for such industrial production tasks. The first consideration is that the environment conditions can be quite variable, ranging from highly dusty environments in ceramics production, to almost open air areas. Illumination can also go from almost dark to daylight in the sun. A laser-based active beacon system has been designed to cope with these conditions. It consists of rotating mirror that deflects the laser beam panoramically, and retro-reflectors (marine tape) to reflect light. A photodiode, an amplifier and signal processing algorithms detect the reflectors orientations relative to the sensor.

A control system has been designed with the aim of providing an easy interface to different steering configurations. Multivariable control theory techniques have been employed to design a control strategy that provides a radius of turn as output, making the vehicle follow the desired path. The path may consist of curves specified as splines as long as they satisfy constraints in the local radius of curvature due to physical limitations in the turns the vehicle can do.

The control system is accompanied of several estimation modules that estimate the motion parameters and other variables needed to provide command signals to the motors, these variables may be different when using different steering configurations.

Since the control system provides a radius of turn, it is independent of the steering mechanism. An adaptation from this output to the motor command signals is proposed for two typical configurations: differential drive and tricycle drive. Both solutions have been implemented and tested in two different prototypes.

Unmanned Ground Vehicle Technology VI, edited by Grant R. Gerhart, Chuck M. Shoemaker, Douglas W. Gage, Proceedings of SPIE Vol. 5422 (SPIE, Bellingham, WA, 2004) · 0277-786X/04/$15 · doi: 10.1117/12.565046

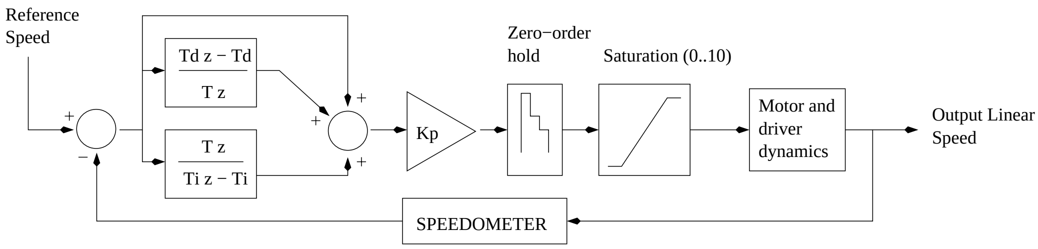

Figure 1. PID linear speed regulator.

2. CONTROL SYSTEM

The purpose of a control system is to provide output signals to control the system from sensory inputs. The theory of feedback control is well known, in the problem of path following the design of a control system consists of finding out which error signals can be available from measurements, deciding which are the best output signals to control the system, identifying the underlying dynamics of the system, and choosing a control scheme.

Our aim is designing a control system for vehicle steering making it independent from the steering mechanism, so that it can be used with different steering solutions. We also pretend to use quite general error signals regardless of the sensors used to measure the vehicle's position.

These considerations lead us to use the vehicle radius of turn and the linear speed as output. The linear speed is a variable that simply has to follow a reference speed fixed by a decision module. Its control can be done with a classical PID regulator assuming the vehicle speed is available from measurements.

Many control schemes for path following have been reported in the literature: behavior-based approaches[1] or adaptive control.[2] We have chosen a classic approach since our goal is not to change control behaviors such as following a path or avoiding an obstacle. In industrial robotics it is not possible nor desirable that a heavy vehicle leaves its path to avoid an obstacle, instead the desired behavior is that the vehicle stops until the obstacle disappears. With this consideration, our control system has only to follow paths, although we intend to specify these paths in a flexible manner, such as generalized curves.

As mentioned, a control of the radius of turn can make the vehicle follow the desired path. This is a two-dimensional problem since the vehicle is moving on a plane. The lateral and orientation offsets with respect to the path to follow are selected as the error signals, and the control scheme has to use a multivariable design technique.

2.1. Linear speed regulator

Fig. 1 shows the block diagram of a PID regulator. The saturation block accounts for real signals to be provided to the motor drivers. The motor (including driver) can be considered a first-order system. In our tests good results have been achieved with $T_d = 0$, $T_I = 0.25$ sec, $K_p = 3$, with a control period of $T = 0.1$ sec.

The speedometer block in Fig. 1 represents any sensor to measure linear speed. In our system the only sensor is laser-based positioning sensor. The linear speed, among other parameters, is estimated from a Kalman filter.

2.2. Path following control

The output of this control scheme is the radius of turn, and its inputs are the lateral offset, d, and orientation error, θ, to the path to follow, Fig. 2. The linear speed is here considered a constant parameter.

For the design of this control system we have used the technique of pole placement in the state space. The state vector is formed by the error signals, $\mathbf{x} = (d \quad \theta)'$. And, for convenience, we consider the inverse of the radius of turn as the input of the states equation: $u = 1/r$. For small θ, and for a path to follow consisting on a straight line (locally), the states equation is linear:

$$\mathbf{x}_{k+1} = \mathbf{A}\mathbf{x}_k + \mathbf{B}u_k \,, \tag{1}$$

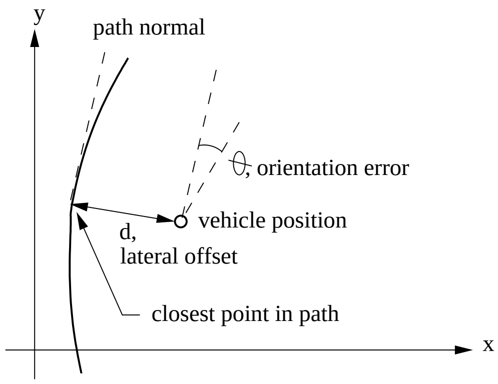

Figure 2. Error signals used in the radius of turn control.

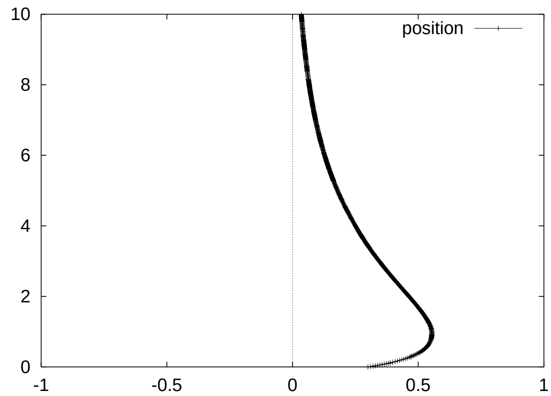

Figure 3. Vehicle positions starting from 30 cm of lateral offset and 45 degrees of orientation error.

where $\mathbf{A} = \begin{pmatrix} 1 & vT \\ 0 & 1 \end{pmatrix}$ and $\mathbf{B} = \begin{pmatrix} 0 \\ vT \end{pmatrix}$, v is the linear speed and T is the control period.

Doing a feedback of the state vector, this control scheme will take the state vector to zero, and so the error signals. It can be seen that the system is controllable since the two-dimensional matrix $(\mathbf{B}|\mathbf{AB})$ is of rank two.

State feedback provides an expression to compute the radius of turn from the error signals:

$$u_k = \mathbf{K}\mathbf{x}_k \,, \tag{2}$$

where $\mathbf{K} = (K_1 \quad K_2)$.

$\mathbf{K}$ is computed from pole placement. If ρ_1 and ρ_2 are the desired poles, the characteristic equation is $\alpha_c(z) = z^2 + \alpha_1 z + \alpha_0$, where $\alpha_1 = -(\rho_1 + \rho_2)$ and $\alpha_0 = \rho_1\rho_2$. The control matrix $\mathbf{K}$ is computed as:

$$\mathbf{K} = (0 \quad 1)\mathbf{F}^{-1}\alpha_c(\mathbf{A}) \,, \tag{3}$$

where $\mathbf{F} = (\mathbf{B}|\mathbf{AB})$ and $\alpha_c(\mathbf{A}) = \mathbf{A}^2 + \alpha_1\mathbf{A} + \alpha_0\mathbf{I}$.

In our experiments we have used real poles since we want to avoid oscillations, with time constants from 5 to 10 seconds. Given a time constant, τ, a real pole value is given by $e^{-T/\tau}$, T being the control period.

The above control scheme has been design from a linear states equation. This linearization is possible if θ is small and the path to follow is locally a line, but the control system will take the vehicle to the line with zero lateral and orientation offset is relatively few time (depending on the time constants). In practice the path to follow can be curved as long as the radius of curvature is smaller than the radius of turn the vehicle can do, which may have physical limitations, although a locally big radius of curvature of the path will make θ big, and this may provide, at some iteration, an impractical radius of turn for the vehicle (too small). In simulations we have seen this control scheme can cope with orientation errors of up to 45 degrees, Fig. 3.

This ability of being able to follow locally linear paths allows us to specify paths to follow not only by lines but also by splines, as long as, for practical purposes, the radius of curvature along the path is always not very low. We use 10 meters of lower bound for local curvature. Higher curvature turns are performed putting the steering wheel in a fixed position during the turn, and not relying on the control system.

3. SYSTEM ARCHITECTURE

As seen, the control system provides commands to the traction and steering motors in form of a linear speed and a radius of turn (the inverse of it). Its inputs are the lateral and orientation offsets and the reference speed.

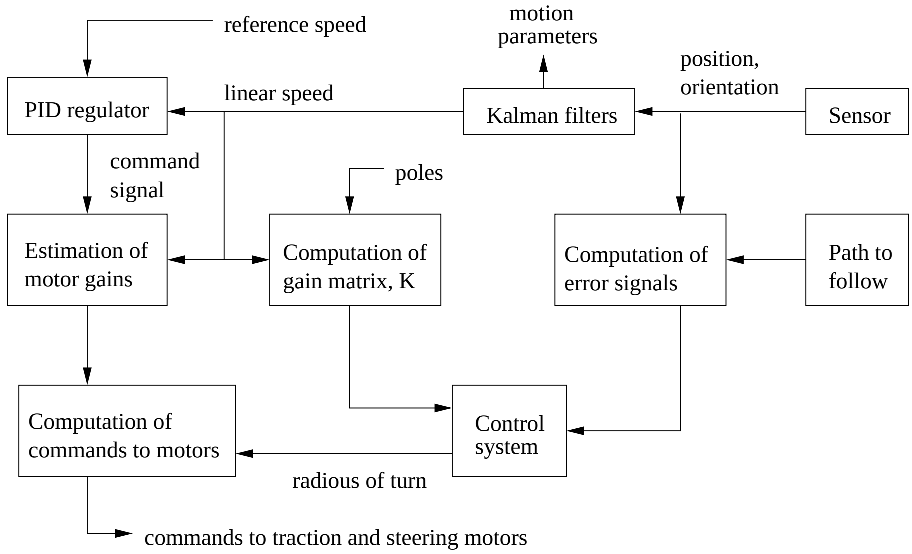

Figure 4. PID linear speed regulator.

To provide the error signals and the reference, there is a need for a higher decision module that fixes the reference speed and the path to follow, and also sensors and estimators to provide the vehicle position and orientation from which the error signals can be computed, and the speed estimated.

To avoid typical error accumulation when relying on wheel encoders and other incremental positioning estimation mechanisms, an absolute positioning system is needed. We use a laser-based positioning system designed by us and built from off-the-shelf optical and electromechanical components. The sensor will be discussed Section 6, but let us assume that an absolute position and orientation are available from the sensor. This information can be used to estimate not only the vehicle linear speed, but also other motion parameters including angular acceleration and radius of turn. Kalman filters and a motion model have been used for this task.

So, the inputs are of two kinds:

- Provided by a decision module: equation of the path to follow, reference speed.
- Provided by sensors and estimators: vehicle position and orientation, vehicle motion parameters.

On the other hand, the outputs of the control system have to be adequate to the steering mechanism, and this may need other parameter estimations since the motor gains are unknown and variable depending on the vehicle load.

With these considerations, the system architecture that integrates the control system with sensor measurements, decision modules, and steering mechanism is shown in Fig. 4. The motion parameters provided by the Kalman filters will be used in the positioning sensor to initiate a new position search, as explained in Section 6. The estimation of the motor gains and the computation of commands to motors depends on the steering mechanism and will be discussed in Section 5.

4. MOTION PARAMETERS ESTIMATION

Motion estimation is usually done with Kalman filters. It can be shown that the Kalman filter is the best linear estimator assuming Gaussian noise in the measurements.[3] Furthermore, Kalman filters provide an iterative solution that takes into account all the previous measurements.

The laser positioning sensor provides a sensor position, (x, y) and orientation, θ. These measurements are performed by triangulation of reflectors positions detected by the sensor. The angles the reflectors have been detected are measured from an incremental encoder but there is an uncertainty in the angle where the center of the reflector lies. Angular uncertainties may produce considerable errors in the estimated sensor position due to the distances, up to 25 meters in our case, the reflectors are from the sensor.

The Kalman filter has the effect of smoothing the measurements, fitting them to a given motion model, and estimating the parameters of this model.

The equations of motion from a given point (x, y, θ) at iteration k, to a new one at iteration $k+1$, given the linear speed, v, and the angular speed, w, are:

$$\left\{ \begin{array}{l} x_{k+1} = x_k + \frac{v_k}{w_k}[-\sin\theta_k(1-\cos w_k T) + \cos\theta_k \sin w_k T] \\ y_{k+1} = y_k + \frac{v_k}{w_k}[\cos\theta_k(1-\cos w_k T) + \sin\theta_k \sin w_k T] \\ \theta_{k+1} = \theta_k + w_k T \end{array} \right. . \tag{4}$$

These equations can be linearized assuming small w ($w \simeq 0$), which is the case in line following:

$$\left\{ \begin{array}{l} x_{k+1} = x_k + v_k T \cos\theta_k = x_k + v_{xk} T \\ y_{k+1} = y_k + v_k T \sin\theta_k = y_k + v_{yk} T \\ \theta_{k+1} = \theta_k + w_k T \end{array} \right. . \tag{5}$$

The three equations can be assigned three different Kalman filters having the same motion model, one for v_x, one for v_y and one for w.

Using a constant acceleration model the three state vectors are: $\mathbf{x} = (x \quad v_x \quad a_x)'$ and $\mathbf{y} = (y \quad v_y \quad a_y)'$ for position x and y, and $\theta = (\theta \quad w \quad \alpha)'$ for orientation θ. a and α stand for linear and angular acceleration.

The transition matrix to get to a new state for this motion model is $\mathbf{F} = \begin{pmatrix} 1 & T & \frac{T^2}{2} \\ 0 & 1 & T \\ 0 & 0 & 1 \end{pmatrix}$, and since we only measure positions, the measurement matrix is $H = (1 \quad 0 \quad 0)$.

The Kalman filter equations are:

$$\text{Prediction} \left\{ \begin{array}{l} \mathbf{x}_{k+1|k} = \mathbf{F}\mathbf{x}_{k|k} \\ \mathbf{P}_{k+1|k} = \mathbf{F}\mathbf{P}_{k|k}\mathbf{F}' + \mathbf{Q} \\ \mathbf{z}_{k+1|k} = \mathbf{H}\mathbf{x}_{k+1|k} \\ \mathbf{S}_{k+1|k} = \mathbf{H}\mathbf{P}_{k+1|k}\mathbf{H}' + \mathbf{R} \end{array} \right. , \tag{6}$$

$$\text{Kalman gain matrix} \left\{ \ \mathbf{W}_{k+1} = \mathbf{P}_{k+1|k}\mathbf{H}'\mathbf{S}_{k+1}^{-1} \right. , \tag{7}$$

$$\text{Update} \left\{ \begin{array}{l} \mathbf{x}_{k+1|k+1} = \mathbf{x}_{k+1|k}\mathbf{W}_{k+1}[\mathbf{z}_{k+1} - \mathbf{z}_{k+1|k}] \\ \mathbf{P}_{k+1|k+1} = [\mathbf{I} - \mathbf{W}_{k+1}\mathbf{H}]\mathbf{P}_{k+1|k}[\mathbf{I} - \mathbf{W}_{k+1}\mathbf{H}]' + \mathbf{W}_{k+1}\mathbf{R}\mathbf{W}_{k+1}' \end{array} \right. , \tag{8}$$

where the subscript $_{k+1|k}$ means estimation at iteration $k+1$ based on k measurements, subscript $_{k+1|k+1}$ means estimation at iteration $k+1$ based on $k+1$ measurements, z_{k+1} is the measured output at iteration $k+1$, $\mathbf{Q}$ is the covariance matrix of the state vector, and $\mathbf{R}$ is the covariance of the measurements, in this case it is a scalar value.

For the Kalman filter design we have to fix $\mathbf{Q}$, $\mathbf{R}$, the first state vector, $\mathbf{x}_{0|0}$, and the first covariance of the state vector, $\mathbf{P}_{0|0}$. These quantities are fixed according to the following expressions: $\mathbf{Q} = \begin{pmatrix} \frac{T^4}{4} & \frac{T^3}{2} & \frac{T^2}{2} \\ \frac{T^3}{2} & T^2 & T \\ \frac{T^2}{2}0 & T & 1 \end{pmatrix} \sigma_a^2$, where σ_a^2 is the variance of the acceleration error, $\mathbf{R} = \sigma_x^2$, variance of the measured position error, $\mathbf{Q} = (x \quad 0 \quad 0)'$, and $\mathbf{P}_{0|0} = cov\{\mathbf{x}_{0|0}\} = \begin{pmatrix} \sigma_x^2 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}$.

Similar expressions stand for the design of the Kalman filters for y position and for angular, θ, position.

5. ADAPTING TO AN STEERING MECHANISM

Since the control system proposed provides a radius of turn, obtaining the commands for the motors implies two steps:

- Writing the motion equations for the desired steering configuration, relating the radius of turn to the motor command signal.
- Dynamically estimating, while the system is working, the unknown parameters that appear in the equations, like motor gains, than can change in time due to different loads or battery charge.

In the following we develop two adaptations of the control system output, which we have implemented and tested, to two different steering mechanisms: differential drive and tricycle drive. Other steering mechanisms, like Ackerman steering, synchro drive or omnidirectional drive, are also possible. Further work can be directed to adapting the control system to these configurations, and estimating their parameters.

5.1. Differential drive

Fig. 5 left shows the variables involved in making a turn of a certain radius r with differential steering. If v is the linear speed of the center of the line base between the two driving wheels, $\delta\theta = \theta_2 - \theta_1$ is the change in heading angle in a small time interval T, such that the wheels rotation is constant in that interval, b is the wheel base, and r is the radius of turn, the displacement of each wheel is:

$$\left\{ \begin{array}{l} d_{right} = \delta\theta(r - \frac{b}{2}) = \frac{vT}{r}(r - \frac{b}{2}) \\ d_{left} = \delta\theta(r + \frac{b}{2}) = \frac{vT}{r}(r + \frac{b}{2}) \end{array} \right. . \tag{9}$$

Assuming the rotation, and so the linear speed of each wheel, is proportional to a command signal (e_{right}, e_{left}):

$$\left\{ \begin{array}{l} v_{right} = \frac{d_{right}}{T} = Ke_{right} \\ v_{left} = \frac{d_{left}}{T} = Ke_{left} \end{array} \right. , \tag{10}$$

where K is the motor gain. On the other hand, for simplicity we consider only deriving the command signals as a central value $\pm$ an increment, where the central value is the one that makes the vehicle accomplish the desired speed (from the PID regulator):

$$\left\{ \begin{array}{l} e_{right} = e - e_{inc} \\ e_{right} = e + e_{inc} \end{array} \right. , \tag{11}$$

$$\left\{ \begin{array}{l} e = \frac{v}{K} \\ e_{inc} = \frac{v}{K}\frac{b}{2}\frac{1}{r} \end{array} \right. . \tag{12}$$

The quantity $\frac{1}{r}$ is directly provided by the control system, v is estimated from the Kalman filters. To estimate the wheel gain, K, we have set up a weighted mean of the last n iterations where the weights decrease in time with a Gaussian form. So this parameter always reflects the recent value of the wheel gain.

5.2. Tricycle drive

Fig. 5 right shows the variables involved in making a turn of a certain radius r with tricycle drive. In tricycle drive the radius of turn is related to the orientation of the steering wheel and the vehicle geometry:

$$r = \frac{l}{\sin\theta}, \tag{13}$$

where l is the length between the front steering wheel and the two rear passive wheels, r is the radius of turn and θ is the vehicle heading angle. The relation between heading angle and radius, $\theta = \sin^{-1}(\frac{l}{r})$ can be easily found, and the problem here is to effectively making the wheel turn exactly at the desired angle, since the relation between wheel angle and motor encoder positions is known from calibration but may have an offset. This offset can be estimated also from measuring the effective radius of turn from two consecutive vehicle positions/orientations at each iteration, and the encoder position at the same iteration. Again a weighted mean with Gaussian weights has been used giving satisfactory results.

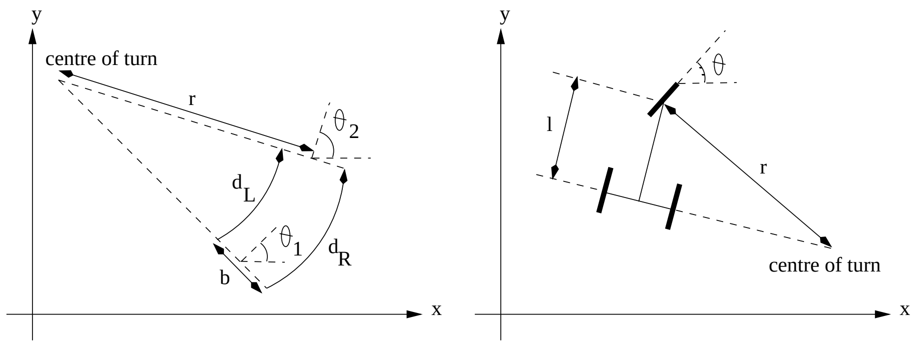

Figure 5. Left: locally constant turn in differential drive. Right: locally constant turn in tricycle drive.

6. POSITIONING SENSOR

A laser-based sensor has been designed for the purpose of estimating the position and orientation of the vehicle. The sensor consists of a rotating laser beam, the beam gets reflected in retro-reflectors disposed at known positions in the work area. The reflected light is filtered and concentrated on a photodiode. The signal induced on the photodiode is amplified and analyzed to detect peaks.

The lase-based local positioning sensor is the only sensor we rely on for our control system. Sonars, shaft encoder-based odometry, range finders, panoramic images, etc, have been also used in other works[4].[5] Nevertheless laser-based landmark detection using retro-reflectors is quite an adequate methodology in dusty and badly illuminated industrial environments.

The angular positions of the peaks are measured from a incremental angular encoder attached to the motor that rotates the beam, obtaining a list of angles. These angles are matched with the expected angles the reflectors are sub-tending from their positions to the vehicle position/orientation. This is done by circularly correlating two signals:

- An *expected signal* consisting on peaks of Gaussian form centered at the angles at which the reflectors would have been detected should the vehicle be in a certain position and orientation. The position and orientation used is the one predicted by the Kalman filters for the present iteration.
- A *sensed signal* consisting on peaks of the same Gaussian form centered at the angles sensed by the sensor.

Position and orientation estimation from landmarks has been usually approached by triangulation. Cohen et al.[6] presented a study on the accuracy of several triangulation methods from three landmarks. When disposing of more that three landmarks, as it is the case in our work, the estimation can to be done from the whole set of detected landmark orientations. Landmark identification implies matching the detected landmarks to real mapped landmarks, avoiding false detections and, afterwards, estimating the position and orientation by a minimization. A work on landmark selection and position estimation can be found elsewhere.[7]

In our work we have used circular correlation as a landmark matching method, and estimation as finding the point of minimum distance to a set of lines. This algorithm has linear computational cost with the number of landmarks detected, it can be run in real time, and provides satisfactory results.

The angle of maximum circular correlation, θ_{max}, provides a first guess of the vehicle orientation. The vehicle orientation is the addition of two angles: the orientation used to build the expected signals plus the angle of maximum correlation. The correlation maximum also provides a matching among detected angles of reflectors and reflectors positions. This matching is obtained by overlapping both signals, but shifting the expected signal with the value of the maximum.

A better estimation is computed by doing a triangulation of all rays coming out from the matched reflectors. The estimation is done by exhaustive search in the orientation angle, shifting it from $\theta_{max} - \theta_{inc}$ to $\theta_{max} + \theta_{inc}$, where θ_{inc} is a small angular increment (a value of 5 deg has been used).

The triangulation is computed at each value of *theta*. It consists of finding the position that has a lower distance to a set of lines. The angular value that provides the lowest estimation error is selected as the estimated vehicle orientation, and the position of lowest distance as the vehicle position.

7. RESULTS

The system discussed in this work has been implemented on two mobile bases, one with differential drive, and one with tricycle drive. As mentioned, only the final interface to adapt the output of the control system to the steering mechanism changes from one solution to the other.

All the system components have been implemented in real time on the *Real-Time Linux* operating system. This operating system provides facilities for writing periodical tasks that execute in hard real time. The signal acquisition, signal analysis, parameter estimation and control loop algorithms have been implemented as periodic real-time tasks, while a graphical user interface to monitor the system runs in the user space of the operating system, in non-real time.

Demonstrations of line following, turns, and curve following have been performed in our installations, providing satisfactory results. The test area is relatively small so far, being a rectangle of 6.5×7.5 meters. The achieved positioning accuracy after trajectories is of the order of 10 cm.

8. CONCLUSIONS

This paper reports a system architecture for the control of an autonomous vehicle to achieve path following, relying only on a laser positioning sensor. The control system provides a general output in terms of a radius of turn, which may be adapted to several steering mechanisms.

Kalman filters have been designed for motion parameters estimation. The motion parameters are used by the control modules to feedback linear speed, and by the positioning system to initiate a search, by means of a correlation of two signals, of the vehicle position and orientation.

The sensor provides a list of angles of detected reflectors. An algorithm for matching the detected angles to known reflectors has been implemented, as well as a method for estimating the vehicle position and orientation, consisting on performing a fine, but quite limited, exhaustive search to look for the orientation that provides a best solution in terms of the triangulation with less fitting error.

The system has been implemented in real time and checked exhaustively. Further work is directed to make an intelligent decision module that provides trajectories to follow, speeds, and load commands to the vehicle, controlling and supervising several load points and several vehicles working at the same time.

REFERENCES

1. E. Bicho and G. Schoner, “The dynamic approach to autonomous robotics demonstrated on a low-level vehicle platform,” *Robotics and Autonomous Systems* **21**, pp. 23–35, 1997.
2. D. Sun, H. Dong, and S. Tso, “tracking stabilization of differencial mobile robots using adaptive synchronized control,” in *Proc. 2002 IEEE International Conference on Robotics and Automation*, pp. 2638–2643, 2002.
3. W. Sorenson, ed., *Kalman filtering: theory and application*, The Institute of Electrical and Electronic Engineers, 1985.
4. F. Matia and A. Jimenez, “Multisensor fusion: an autonomous mobile robot,” *Journal of Intelligent and Robotic Systems* **22**, pp. 129–141, 1998.
5. L. Moreno and E. Lapena, “Path quality measures for sensor-based motion planning,” *Robotics and Autonomous Systems* **44**, pp. 131–150, 2003.
6. C. Cohen and F. Koss, “A comprehensive study of three object triangulation,” in *Proc. 1993 SPIE Conference on Mobile Robots*, pp. 95–106, 1992.
7. M. Betke and L. Gurvits, “Mobile robot localization using landmarks,” *IEEE Transactions on Robotics and Automation* **13**(2), pp. 251–263, 1997.