

CONTROLLER SYNTHESIS APPLIED TO AUTOMATIC GUIDED VEHICLES

Philippe Martinet, Djamel Khadraoui,
Christian Thibaud, Jean Gallice

*Laboratoire des Sciences et Matériaux pour l'Electronique,
et d'Automatique.*

*Université Blaise Pascal de Clermont-Ferrand,
U.M.R. 6602 du C.N.R.S., F-63177 Aubière Cedex, France
E-Mail: Philippe.Martinet@lasmea.univ-bpclermont.fr*

Abstract: We have been interested in Automatic Guided Vehicles (AGV) for several years. In this paper, we synthesize controllers for AGV applications. Particularly, we are interested in road following and direction change tasks, and in analyzing the influence of roll and pitch perturbations on vehicle behaviour. We use the bicycle as the kinematic vehicle model, and we choose the white band position of the road as the sensor signal. We define an interaction between the camera, which is mounted inside the vehicle, and the white band detected in the image space. Using this kind of interaction, we present how to use a pole assignment technique to solve the servoing task. We show the simulation and experimental results (1/10 scale demonstrator) with and without perturbations.

Keywords: Visual servoing, robot control, mobile robot, vehicles, modelling, vision

1. INTRODUCTION

In the realm of intelligent systems for highways, development of AGV is necessary to enable vehicles to drive automatically along the road. In fact, the requirement is for a controller that can maintain the position and the orientation of the vehicle with respect to the centre of the road and/or apply changes of direction. The problem of vehicle control using a camera has been given considerable attention by many authors (Dickmanns and Zapp, 1987; Kehtarnavaz *et al.*, 1991; Raymond and Chaouchi, 1994; Wallace *et al.*, 1986; Waxman *et al.*, 1987). The work described in (Jurie *et al.*, 1992; Jurie *et al.*, 1994) is among the most notable in lateral control using monocular vision. It consists of the reconstruction of the road using the 2D visual information extracted from the image processing system (Chapuis *et al.*, 1995). In recent years, the integration of computer vision in robotics has steadily progressed,

from the early “look and move” systems, to current systems in which visual feedback is incorporated directly into the control loop. These techniques of vision based control are used to control holonomic robots in different domains (Feddema and Mitchell, 1989; Khadraoui *et al.*, 1996; Papanikolopoulos *et al.*, 1991; Papanikolopoulos *et al.*, 1993). The principle of this approach is described in figure 1.

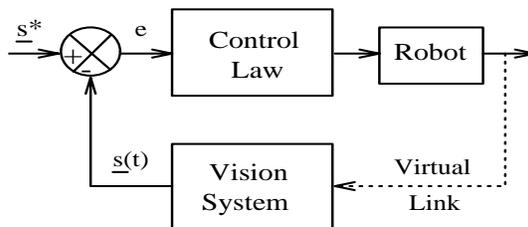


Fig. 1. Visual servoing scheme.

It is based on the task function approach (Samson *et al.*, 1991), and many people have developed this concept applied to visual sensors (Chaumette, 1990; Espiau *et al.*, 1992; Hutchinson *et al.*, 1996). There are still few applications in mobile robots using this kind of approach. The main difficulty is due to the presence of nonholonomic mechanical connections which limit robot movements (Pissard-Gibollet and Rives, 1991).

In this paper, we propose a new technique with a visual servoing approach, in which control incorporates the visual feedback directly (Khadraoui *et al.*, 1995). In other words, this is specified in terms of regulation in the image frame of the camera. Our application involves controlling the lateral road position of a vehicle following the motorway white line. A complete 2D model of both the vehicle and the scene is then essential. It takes into account the visual features of the scene and the modelling of the vehicle.

The main purpose of this study is the development of a new lateral control algorithm. We propose a new control model, based on state space representation, where the elements of the state vector are represented by the parameters of the scene, extracted by vision. This approach was experimented with a 1/10 scale demonstrator.

2. MODELLING ASPECT

Before synthesizing the control laws, it is necessary to obtain the model of the vehicle and the one which expresses the interaction between the sensor and the environment.

2.1 Modelling the vehicle

It is useful to approximate the kinematic of the steering mechanism by assuming that the two front wheels turn slightly differentially. Then, the instantaneous center of rotation can be determined purely by kinematic means. This amounts to assuming that the steering mechanism is the same as that of a bicycle. Let the angular velocity vector directed along z axis be called $\dot{\psi}$ and the linear one directed along x axis called \dot{x} .

Orientation equation: Using the bicycle model approximation (see figure 2 (a)), the steering angle δ and the radius of curvature r are related to the wheel base L by:

$$\tan \delta = \frac{L}{r} \quad (1)$$

In figure 2 (b), we show a small portion of a circle ΔS representing the trajectory to be followed by

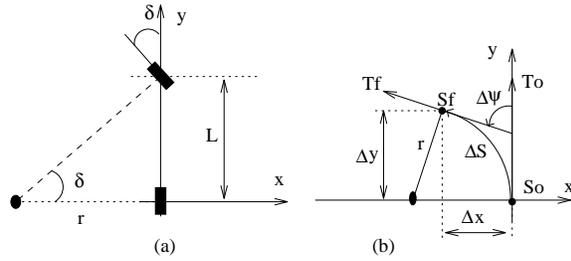


Fig. 2. Bicycle model.

the vehicle. We assume that it moves with small displacements between initial curvilinear abscissa S_0 and a final one S_f such that:

$$\frac{1}{r} = \lim_{\Delta s \rightarrow 0} \frac{\Delta \psi}{\Delta S} = \frac{d\psi}{dS} = \frac{d\psi}{dt} \frac{dt}{dS} \quad (2)$$

where ψ represents the orientation of the vehicle. The dot of S includes the longitudinal and lateral velocities along the y and x axes respectively. In fact, the rotation rate is obtained as:

$$\dot{\psi} = \frac{\tan \delta}{L} \sqrt{\dot{x}^2 + \dot{y}^2} \quad (3)$$

Lateral position equation: In order to construct this equation, we treat the translational motion assuming that the vehicle moves with small displacements between t and $t + \Delta t$. In the case of a uniform movement during a lapse of time Δt , the vehicle moves through distance $d = V\Delta t$ taking V as a constant longitudinal velocity (see figure 3).

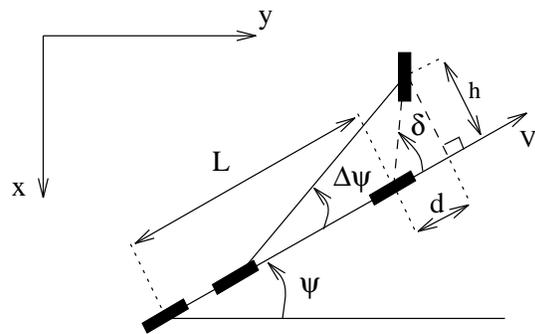


Fig. 3. Kinematic modelling of the vehicle.

We express:

$$\begin{cases} S\dot{\psi} = - \lim_{\Delta t \rightarrow 0} \frac{x_{t+\Delta t} - x_t}{V\Delta t} = - \frac{\dot{x}}{V} \\ C\dot{\psi} = \frac{\dot{y}}{V} \end{cases} \quad (4)$$

In these expressions, C and S represent the trigonometric functions cosine and sine.

The approximation to small angles gives us the

relation between the differential of the lateral coordinate x and the lateral deviation ψ with regard to δ , expressed as follows:

$$\begin{cases} \dot{x} = -V\psi \\ \dot{\psi} = \frac{V}{L}\delta \end{cases} \quad (5)$$

2.2 Modelling the scene

The present section shows how to write the equation of the projected line in the image plane, using perspective projection. The scene consists of a 3D line and its projected image is represented by a 2D line. Figure 4 shows the frames used in order to establish this relation.

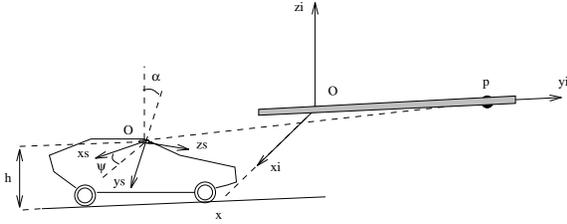


Fig. 4. Perspective projection of a 3D line.

We use:

- $R_i = (O, x_i, y_i, z_i)$ as the frame attached to the 3D line
- $R_s = (O, x_s, y_s, z_s)$ as the sensor frame fixed to the camera

We take into account:

- h the camera height
- α the inclination angle of the camera
- ψ the orientation of the vehicle

Any 3D point $p_i = (x_i, y_i, z_i, 1)^T$ related to the workspace can be represented by its projection in the image frame $p_p = (x_p, y_p, z_p)^T$ by the relation:

$$p_p = M p_i \quad (6)$$

The matrix M represents the homogeneous calibration matrix of the camera expressed in the frame R_i such as:

$$M = P'_c C'_u C_a R_2^{-1} R_1^{-1} T^{-1} \quad (7)$$

where:

- $P'_c C'_u C_a$ translates $p_s = (x_s, y_s, z_s)^T$ in p_p . $P'_c C'_u$ takes into account the intrinsic parameters of the camera ($f_x = f e_x, f_y = f e_y$), and C_a realizes an exchange coordinates frame.
- T, R_1 and R_2 characterize the extrinsic parameters of the camera. T takes into account the two translations of the camera: the height h and the lateral position x . R_1 and R_2

represent the lateral orientation ψ and the inclination α of the camera respectively.

The expressions of the different matrices used are given as follows:

$$P'_c C'_u = \begin{bmatrix} f_x & 0 & 0 & 0 \\ 0 & f_y & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}, \quad C_a = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

$$T = \begin{bmatrix} 1 & 0 & 0 & x \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & h \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad R_1 = \begin{bmatrix} C\psi & -S\psi & 0 & 0 \\ S\psi & C\psi & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$R_2 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & C\alpha & -S\alpha & 0 \\ 0 & S\alpha & C\alpha & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

We note that the second translation and the third rotation are considered as null, and that x (lateral position) and ψ (orientation) are the system variables.

Developing relation 7, we obtain the following expression of M :

$$\begin{bmatrix} f_x C\psi & f_x S\psi & 0 & f_x x C\psi \\ -f_y S\alpha S\psi & f_y S\alpha C\psi & -f_y C\alpha & -f_y x S\alpha S\psi + f_y h C\alpha \\ -C\alpha S\psi & C\alpha C\psi & S\alpha & -x C\alpha S\psi + h S\alpha \end{bmatrix}$$

The pixel coordinates $P = (X = \frac{x_p}{z_p}, Y = \frac{y_p}{z_p})^T$ associated with each point of the 3D line (with $x_i = z_i = 0$), are expressed by:

$$\begin{cases} X = f_x \frac{y_i S\psi + x C\psi}{y_i C\alpha C\psi - x C\alpha S\psi + h S\alpha} \\ Y = f_y \frac{-y_i S\alpha C\psi - x S\alpha S\psi + h C\alpha}{y_i C\alpha C\psi - x C\alpha S\psi + h S\alpha} \end{cases} \quad (8)$$

Eliminating y_i from the equations 8, we obtain:

$$X = \frac{f_x}{f_y} \left[\frac{x C\alpha - h S\psi S\alpha}{h C\psi} \right] Y + f_x \left[\frac{x S\alpha + h S\psi C\alpha}{h C\psi} \right] \quad (9)$$

Considering that α and ψ are small (< 10 deg) and if we neglect the product term $\alpha\psi$, the second order *Taylor* approximation gives us a new expression of X :

$$\frac{f_x x}{f_y h} Y + f_x \left(\frac{x\alpha}{h} + \psi \right) + O(\psi^2) + O(\alpha^2) \quad (10)$$

The equation of the line expressed in the image frame is given by the following relation:

$$X = aY + b \quad (11)$$

where (a, b) are the line parameters expressed by:

$$\begin{cases} a = \frac{f_x x}{f_y h} & = \mu_1 x \\ b = f_x \left(\frac{\alpha x}{h} + \psi \right) & = \mu_2 x + \mu_3 \psi \end{cases} \quad (12)$$

We express the lateral position x and the orientation of the vehicle ψ in order to define an interaction relation between the 3D position of the vehicle and the 2D parameters of the line in the image plane. We have:

$$\begin{cases} \dot{x} = \frac{h f_y \dot{a}}{f_x} = \xi_1 \dot{a} \\ \dot{\psi} = -\frac{\alpha f_y \dot{a}}{f_x} + \frac{1}{f_x} \dot{b} = \xi_2 \dot{a} + \xi_3 \dot{b} \end{cases} \quad (13)$$

with: $f_x = 1300pu$, $f_y = 1911pu$, $V = 20km/h$, $L = 0.3m$ and $h = 0.12m$ (in our 1/10 scale demonstrator).

3. CONTROL ASPECT

In this section, we show how to synthesize controllers with a pole assignment technique. First, we establish the state equation of the system, and we define a controller with different characteristics. We verify by simulation the system behaviour without perturbations. We then introduce some perturbations, and analyse the simulation results. Secondly, we modify the control law by appending an integrator and present simulation results.

3.1 Pole assignment approach

3.1.1. Controller design

Here, we present the application of the pole assignment technique when the state model is expressed directly in the sensor space. In our case, the sensor space is the image plane. The controller design is based on the kinematic model of the vehicle. We use the (a, b) parameters of the 2D line in the image plane as the state vector. We drive the vehicle with the action on the wheel angle δ . We choose b as the output parameter of the system and use the results of the vehicle and scene modellings to obtain the following equation:

$$\begin{cases} -V\psi & = \xi_1 \dot{a} \\ (V/L)\delta & = \xi_2 \dot{a} + \xi_3 \dot{b} \end{cases} \quad (14)$$

The state vector, denoted by $\underline{s} = (a, b)^T$, is equal to the sensor signal vector in the state space representation. Developing, we have the following state model of the system:

$$\begin{cases} \dot{\underline{s}} = A\underline{s} + B\delta \\ b = C\underline{s} \end{cases} \quad (15)$$

$$\text{with : } A = \begin{bmatrix} -\frac{V\xi_2}{\xi_1} & -\frac{V\xi_3}{\xi_1} \\ \frac{V\xi_2^2}{\xi_1\xi_3} & \frac{V\xi_2}{\xi_1} \end{bmatrix}, B = \begin{bmatrix} 0 \\ \frac{V}{L\xi_3} \end{bmatrix} \\ C = [0, 1]$$

The visual servoing scheme is then as shown in the figure 5.

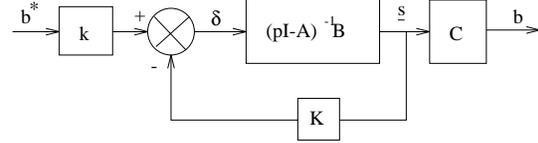


Fig. 5. Visual servoing scheme with a pole assignment

Finally, we can express the control law by the following relation:

$$\delta = -k_1 a - k_2 b + k b^* \quad (16)$$

where $K = [k_1, k_2]$ and k are the gains of the control law obtained by identifying the system to a second order system characterized by ω_0 and ξ . For these control gains, we obtain:

$$\begin{cases} k_1 = \frac{L\omega_0 (2\xi_2 \xi V - \xi_1 \omega_0)}{V^2} \\ k_2 = \frac{2L\xi_3 \xi \omega_0}{V} \\ k = \frac{\xi_1 L\xi_3 \omega_0^2}{V^2 \xi_2} \end{cases} \quad (17)$$

We note that, as expressed by these relations, the higher the velocity V , the smaller are the gains.

3.1.2. Simulation results

To validate this control law, we use a simulator developed with Matlab. Figure 6 shows the visual servoing scheme used in this simulator.

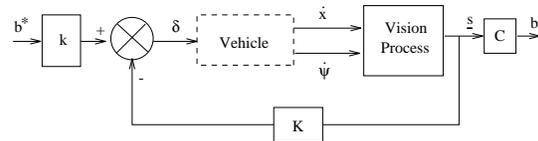


Fig. 6. Visual servoing scheme of the simulator

We use the kinematic model of the vehicle to simulate the behaviour of the vehicle, and the perspective projection relation to obtain the sensor signal $\underline{s} = (a, b)^T$. The first results (see figure 7) illustrate the output behaviour of the system corresponding to an input value $b^* = 100$ pixels. We take into account a data flow latency (three

sample periods) in all simulation tests. This was identified on our experimental site.

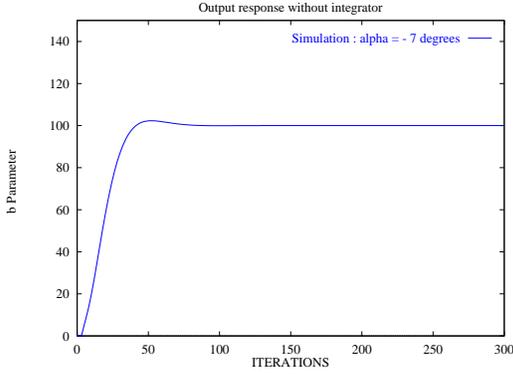


Fig. 7. b without perturbations ($\alpha = -7$ degrees)

We chose $\omega_0 = 2rd/s$ and $\xi = 0.9$ in order to fix the behaviour of the system. In this case, we have no perturbations (α is fixed at -7 degrees).

The second set of results takes into account a perturbed angle α . We obtain the following response in b (see figure 8):

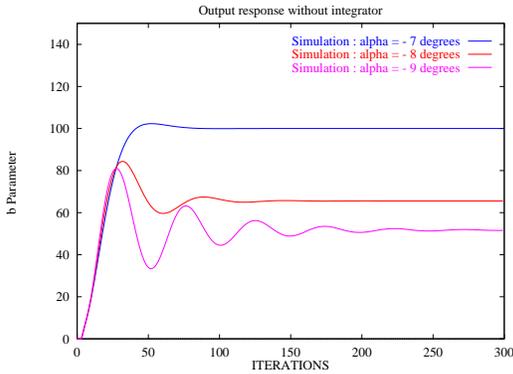


Fig. 8. b with perturbations ($\alpha = -7,-8,-9$)

We can see a static error in b , when we introduce the perturbations, and some oscillations appear when the α angle increases. In this case, ξ slows down from 0.9 to 0.6 and increases the overtaking, but the main contribution to the oscillations is due to the data flow latency.

In the next section, we express the static error in order to analyze the simulation and experimental results.

3.2 Closed loop static error estimation

In this part, we analyse the behaviour of the vehicle when the extrinsic parameters of the camera are perturbed. In this case, equations 13 shows us that only ξ_1 and ξ_2 are affected by this kind of perturbations.

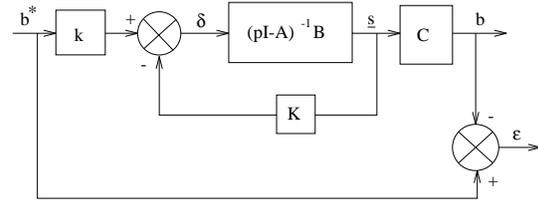


Fig. 9. Static error

Considering the state equations 15 and taking into account parameters perturbations, we have:

$$A = \begin{bmatrix} -\frac{V(\xi_2 + \Delta\xi_2)}{\xi_1 + \Delta\xi_1} & -\frac{V\xi_3}{\xi_1 + \Delta\xi_1} \\ \frac{V(\xi_2 + \Delta\xi_2)^2}{(\xi_1 + \Delta\xi_1)\xi_3} & \frac{V(\xi_2 + \Delta\xi_2)}{\xi_1 + \Delta\xi_1} \end{bmatrix}$$

Figure 9 represents the general visual servoing scheme, using the pole assignment approach. In this case, we can establish the general expression of the static error ϵ_∞ by using the Laplace transform by:

$$\epsilon(p) = b^*(p) - b(p) \quad (18)$$

with:

$$b(p) = C [pI - (A - BK)]^{-1} B k b^*(p) \quad (19)$$

Hence we obtain:

$$\epsilon_\infty = \left[1 + C (A - BK)^{-1} B k \right] b_\infty^* \quad (20)$$

After some developments and approximations, we can write the final relation of the static error as follows:

$$\epsilon_\infty = \left[1 - \frac{1 + \frac{\Delta\xi_2}{\xi_2}}{1 + \frac{2V\xi_2}{\omega_0} \frac{\xi_2}{\xi_1} \frac{\Delta\xi_2}{\xi_2}} \right] b_\infty^* \quad (21)$$

The static error becomes null, if the following expression is verified:

$$\frac{\Delta\xi_2}{\xi_2} = \frac{\Delta\alpha}{\alpha} = 0 \quad (22)$$

We observe that, in the absence of perturbations in the α parameter, we have no static error. When α is different from the reference value, we obtain an error which confirms all the simulation results. In the next section, we show the way to reduce this static error.

3.3 Pole assignment with integrator

In this section, we introduce an integrator into the control law in order to eliminate the static error in case of perturbations.

3.3.1. Controller design

The visual servoing scheme is presented by the figure 10 .

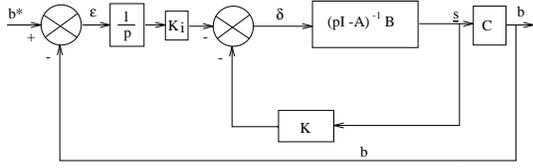


Fig. 10. Visual servoing scheme with integrator

In this case, we can express the control law by the relation:

$$\delta = -k_1 a - k_2 b - K_i \int (b^* - b) dt \quad (23)$$

where k_1 , k_2 and K_i are the gains of the control law obtained by identifying the system to a third order system characterized by the following characteristic equation: $(p^2 + 2\xi\omega_0 p + \omega_0^2)(p + \xi\omega_0)$. The control law gains are given by:

$$\begin{cases} k_1 = -\frac{L\omega_0 (2\xi_1 \omega_0 \xi^2 - 3\xi_2 \xi V + \xi_1 \omega_0)}{V^2} \\ \quad + \frac{L\xi_1^2 \omega_0^3 \xi}{V^3 \xi_2} \\ k_2 = \frac{3L\xi_3 \xi \omega_0}{V} \\ K_i = -\frac{\xi_1 L\xi_3 \omega_0^3 \xi}{V^2 \xi_2} \end{cases} \quad (24)$$

3.3.2. Simulation results

In the first simulation, we have no perturbations. Figure 11 illustrates the fact that the response time is correct.

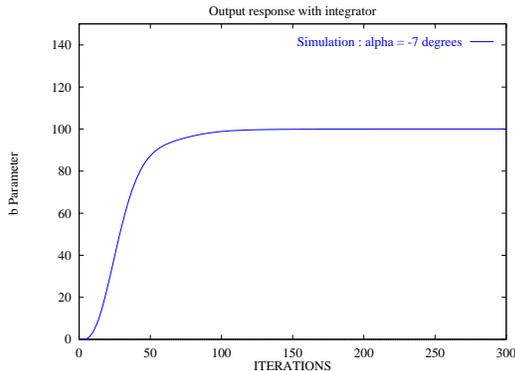


Fig. 11. b without perturbations ($\alpha = -7$)

Secondly, we introduce some perturbations in the α angle. In fact, no static error persists during servoing, but some oscillations and problems of stability appear when α is far from the reference value. Figure 12 shows these results.

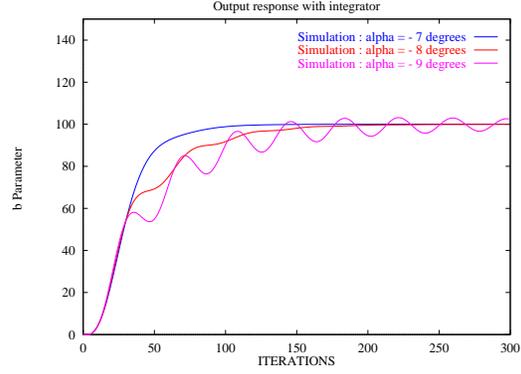


Fig. 12. b with perturbations ($\alpha = -7, -8, -9$)

4. EXPERIMENTAL RESULTS

In this section, we present different experimental results. The CCD camera is embedded on the end effector of a cartesian robot and is connected to the vision system WINDIS.

4.1 Experimental site

To validate our approach, we programmed the control laws on our experimental robotic platform. It is composed of a cartesian robot with 6 degrees of freedom (built by the firm AFMA Robot) and the parallel vision system WINDIS (Martinet *et al.*, 1991; Rives *et al.*, 1993). This whole platform is controlled by a VME system, and can be programmed in C language under the VxWorks real time operating system.

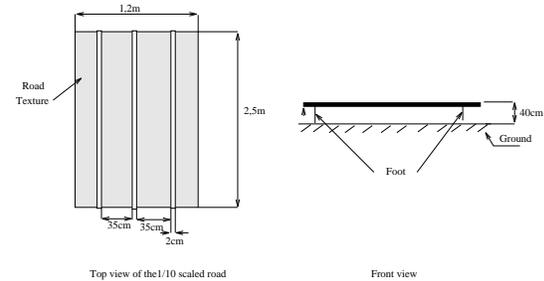


Fig. 13. Road on a 1/10 scale.

Figure 13 presents the road built to a 1/10 scale. This road comprises three white lines.

Using this kind of testbed allows us to use the visual servoing scheme illustrated in figure 14.

4.2 The vision system

Windis architecture comprises three basic modules (see figure 15).

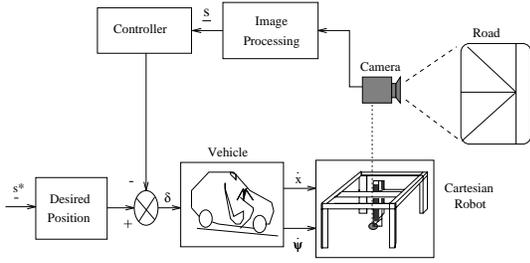


Fig. 14. Visual servoing scheme

The three basic modules are: the Window Distributor Subsystem, the WINPROC Window Processing Subsystem and the WINMAN Window Manager Subsystem.

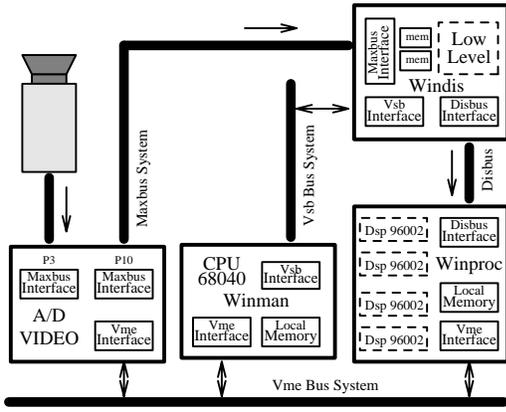


Fig. 15. Overview of Windis architecture.

For each level, we introduced parallelism allowing us to reach video rate for most of the application tasks. The management of such a system is a tricky job and requires the use of a real time operating system. For ease of use, we choose to develop the whole system under VxWorks.

The Windows Distributor extracts points from regions of interest and sends them through the bus *Disbus*. The DSP module (intermediate level processing) computes the (a, b) parameters of the line from these selected points at video rate. In this implementation, we have identified a data flow latency of three sample periods.

4.3 Experimental results comments

First, we experiment the control law with and without perturbations. As in simulation, we obtain a static error and some oscillations (see figure 16). In this experiment, we can verify the values of static error for different values of the α parameter. For

$$\begin{aligned} \alpha = -8 \text{ we have } \epsilon_{\infty} &= 35/34 \\ \alpha = -9 \text{ we have } \epsilon_{\infty} &= 49/48 \\ (\text{measured value}/ \text{computed value}). \end{aligned}$$

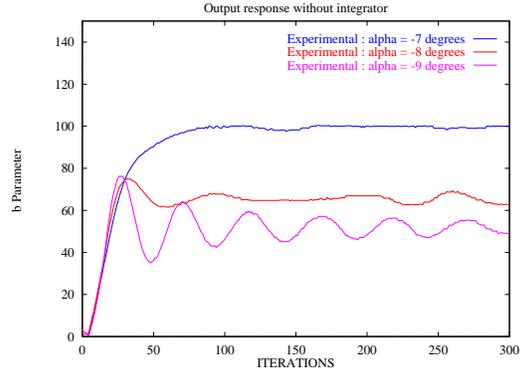


Fig. 16. b parameter: $\alpha = -7, -8, -9$

Secondly, we modify the control law by introducing an integrator in the visual servoing scheme. Figure 17 shows the results. We obtain the reduction of the static error in case of perturbation, but some oscillations and instability appears when α is distant from the reference value of α (-7 degrees).

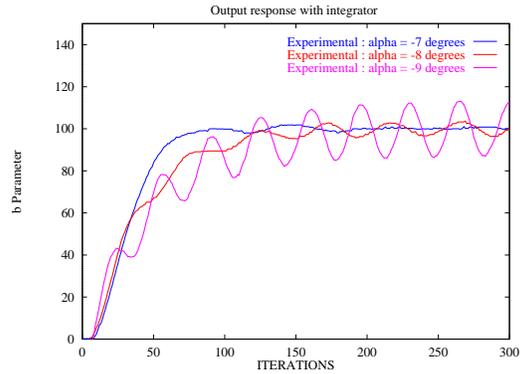


Fig. 17. b parameter: $\alpha = -7, -8, -9$

5. CONCLUSION AND FUTURE WORK

Controllers based on a visual servoing approach, have been developed in this paper. We designed a controller with a pole assignment technique directly in the image space. After modelling the vehicle and the scene, we obtained equations which can be used to write the state model of the system. Visual servoing is performed well when there are no perturbations. When perturbations occur, a static error and oscillations appear. By introducing an integrator into the visual servoing scheme, we suppress the static error but amplify the oscillation problem.

In the future, we will investigate a robust control approach, which means defining a controller which must be robust with regard to wide variations of parameter modelling. The choice of b as the output parameter of the system does not permit the control of the lateral position of the vehicle precisely when the perturbations appear, but insure a heading control. We look in the future how to control the system with the a parameter under

perturbations. We assume that experimentation on a real vehicle will be necessary to validate all of the primary results obtained in our laboratory.

6. REFERENCES

- Chapuis, R., A. Potelle, J.L. Brame and F. Chausse (1995). Real-time vehicle trajectory supervision on the highway. *The International Journal of Robotics Research* **14**(6), 531-542.
- Chaumette, F. (1990). La relation vision commande: théorie et application à des tâches robotiques. PhD thesis. IRISA/INRIA. Rennes, France.
- Dickmanns, E.D. and A. Zapp (1987). Autonomous high speed road vehicle guidance by computer. In: *Proceedings of 10th IFAC World Congress*.
- Espiau, B., F. Chaumette and P. Rives (1992). A new approach to visual servoing in robotics. *IEEE Transactions on Robotics and Automation* **8**(3), 313-326.
- Feddema, J. T. and O. R. Mitchell (1989). Vision-guided servoing with feature-based trajectory generation. *IEEE Transactions on Robotics and Automation* **5**(5), 691-700.
- Hutchinson, S., G.D. Hager and P. Corke (1996). A tutorial on visual servo control. *IEEE Transactions on Robotics and Automation* **12**(5), 651-670.
- Jurie, F., P. Rives, J. Gallice and J.L. Brame (1992). A vision based control approach to high speed automatic vehicle guidance. In: *Proceedings of IARP Workshop on Machine Vision Applications*. Tokyo, Japan. pp. 329-333.
- Jurie, F., P. Rives, J. Gallice and J.L. Brame (1994). High-speed vehicle guidance based on vision. *Control Engineering Practice* **2**(2), 287-297.
- Kehtarnavaz, N., N.C. Grisworld and J.S. Lee (1991). Visual control for an autonomous vehicle (bart)-the vehicle following problem. *IEEE Transactions on Vehicular Technology* **40**(3), 654-662.
- Khadraoui, D., G. Motyl, P. Martinet, J. Gallice and F. Chaumette (1996). Visual servoing in robotics scheme using a camera/laser-stripe sensor. *IEEE Transactions on Robotics and Automation* **12**(5), 743-749.
- Khadraoui, D., P. Martinet and J. Gallice (1995). Linear control of high speed vehicle in image space. In: *Proceedings of Second International Conference on Industrial Automation*. Vol. 2. IAIA. Nancy, France. pp. 517-522.
- Martinet, P., P. Rives, P. Fickinger and J.J. Borrelly (1991). Parallel architecture for visual servoing applications. In: *Proceedings of the Workshop on Computer Architecture for Machine Perception*. Paris, France. pp. 407-418.
- Papanikolopoulos, N., P.K. Khosla and T. Kanade (1991). Vision and control techniques for robotic visual tracking. In: *Proceedings of the IEEE International Conference on Robotics and Automation*. Sacramento - USA.
- Papanikolopoulos, N., P.K. Khosla and T. Kanade (1993). Visual tracking of a moving target by a camera mounted on a robot: a combination of control and vision. *IEEE Transactions on Robotics and Automation* **9**(1), 14-35.
- Pissard-Gibollet, R. and P. Rives (1991). Asservissement visuel appliqué à un robot mobile: état de l'art et modélisation cinématique. Technical Report 1577. Rapport de recherche INRIA.
- Raymond, H.B. and A. Chaouchi (1994). Robust lateral control of highway vehicles. In: *Proceedings of the Intelligent Vehicles '94 Symposium*. Paris, France.
- Rives, P., J.L. Borrelly, J. Gallice and P. Martinet (1993). A versatile parallel architecture for visual servoing applications. In: *Proceedings of the Workshop on Computer Architecture for Machine Perception*. News Orleans, USA. pp. 400-409.
- Samson, C., M. Le Borgne and B. Espiau (1991). *Robot Control: The task function approach*. ISBN 0-19-8538057. Oxford University Press.
- Wallace, R., K. Matsuzak, Y. Goto, J. Crisman, J. Webb and T. Kanade (1986). Progress in robot road following. In: *Proceedings of the IEEE International Conference on Robotics and Automation*.
- Waxman, A.M., J. LeMoigne, L.S. Davis, B. Srinivasan, T.R. Kushner, E. Liang and T. Sidalgaiah (1987). A visual navigation system for autonomous land vehicles. *IEEE transactions on Robotics and Automation* **3**(2), 124-141.