A Fast Vision-based Road Following Strategy Applied to the Control of Aerial Robots

GERALDO F. SILVEIRA[1], J. REGINALDO H. CARVALHO[2], MARCONI K. MADRID[1],
PATRICK RIVES[3], SAMUEL S. BUENO[2]

[1]UNICAMP – PO Box 6101, 13081-970, Campinas-SP, Brazil.
{silveira,madrid}@dsce.fee.unicamp.br
[2]Robotics and Computer Vision Laboratory/ITI – PO Box 6162, 13081-970, Campinas-SP, Brazil.
{First_name.Last_name}@iti.br
[3]INRIA Sophia-Antipolis – 2004 Route des Lucioles, 06565 Valbonne Cedex, France
Patrick.Rives@sophia.inria.fr

Abstract. The utilization of the vision as a feedback sensor in closed-loop control schemes is of great interest, mainly due to the high density of information revealed by images. In this work, we present a method to perform road following tracking by aerial unmanned vehicles (AUV) based on visual input. Difficulties arise from the non-holonomic constraints of the AUV moving in 3D. The problems are overcome using a visual servoing approach based on an Interaction Matrix. Simulation results validating the strategy are shown at the end of the paper.

1 Introduction

Vision as a feedback sensor in closed-loop control schemes is of great interest, mainly due to the high density of information revealed by images. Thus, it constitutes an alternative to increase the overall accuracy of the system, which is an important concern in any application.

The approach proposed in [3] is applied to the visual servoing of a class of AUV (Airship, figure 1) to perform road following. The solution is based on the so-called Interaction Matrix (IM) [10], whose purpose is to represent the robotic Jacobian within the image plane, using certain geometric primitives as reference images. Then, the velocities of the image features parameters are used *directly* to compute control signals for driving the robot actuators without any state reconstruction or camera calibration schemes. In such way, we reduce the computational effort, increasing the tracking speed and the robustness of the system.

Research on the utilization of aerial unmanned vehicles (AUV) appeared in the last decade due to the huge list of potential applications [7], such as surveillance, inspection of power lines and pipelines, mineral and archaeological site exploration and prospection, law enforcement and telecommunication relay systems. A new and special attention has been given to the use of AUV's in environmental applications, biodiversity and climate research, including sensoring and monitoring of forests, national parks and ecological sites, land use survey, livestock inventory, air composition and pollution measurement above cities and industrial sites, characterization of plants and animals, among others. For these kind of tasks, airships out-rank helicopters and airplanes as the ideal platform, mainly because their aerostatic lift makes them less intrusive, noiseless, capable of hovering and able to fly during longer periods.

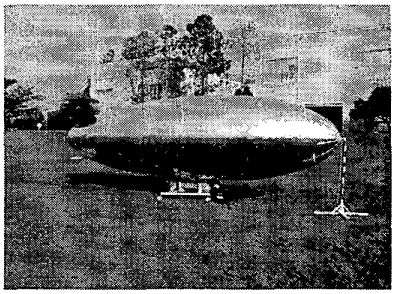

Figure 1: Aerial Unmanned Robot.

The intended autonomy level of robotic systems depends strongly on the processing and interpretation capability of all sensing information, despite the disturbances found in real world applications such as wind and gust.

Among all control problems potentially solvable by vision, some are particularly interesting for AUV's such as automatic docking, hovering, and river, roads, pipes and power lines tracking. Recently, vision-based methodologies designed to navigate robotic vehicles have been targets of researchers. The reader may refer to [6], [13] and [9] for other examples of visual servo control.

As the basis for the development of control and navigation strategies, a 6 DOF model, SIMULINK-based control system development environment was used for design and validation purposes. In order to visualize and to represent the world where the vehicle is flying around, it is used a visualizator (see figure 2) based on Mesa (a freeware implementation of the openGL language). The systems communicate with each other through TCP/IP sockets, increasing flexibility.

1530-1834/01 $10.00 © 2001 IEEE

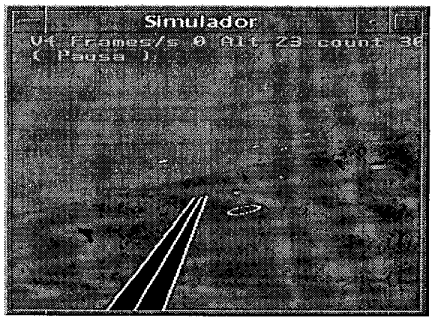

Figure 2: Simulator developed. Cockpit view.

This paper is organized as follows: Section 2 describes the task specification. In Section 3, the implementation of vision aspects used in this work is presented. In Section 4 the control aspects are discussed and in Section 5 the results are shown, demonstrating the feasibility of the system. Finally, in Section 6 some remarks and future work are stated to conclude the paper.

2 Problem Formulation

The main objective of this paper is to develop a pure vision-based control application which is both useful and feasible using off-the-shelf components.

Visual servo control is established when data from an image is fedback to the control in a closed-loop scheme [4]. In some cases, the pixel information is processed to rebuild the tri-dimensional configuration of the robot and these variables are used as control input. In other methodologies the pixel position and its intensity are used directly in the control algorithm computations. In any case, if it is possible to merge everything in a closed-loop control block, having data from image as input and as output the robot control signals, then one has a visual servo control application.

The task addressed here can be stated as: given a set of lines, i.e., a road, the robotic airship has to follow it with a established altitude and longitudinal speed. The problem may also be formulated in the image plane as regulating the image so that the central line is intended to be vertically centered with both lateral lines lying simetrically with a certain inclination. The number of lines used here (three) is the minimum number to perform such a task avoiding the redundancy that appears due to the projection of the actual target in the image plane.

Difficulty arises since the vehicle moves in 3D and such an image can be distorted according to a fairly general set of deformations, when compared to those usually considered in robotics and computer vision.

Real world applications of visual servoing techniques can be divided into vision aspects and control aspects. Concerning vision, the tasks include image acquisition and segmentation, feature tracking and computation of feature parameters. Concerning control, one has the configuration and speed transformations among all frames assigned to the system, kinematic and/or dynamic modeling, Jacobian computation and controller design. This work will focus on the vision aspects.

3 Vision Aspects

3.1 Preliminaries

Let us represent the camera in a perspective projection model. This model defines a local differential mapping between the configuration and the sensor output spaces ($s : SE_3 \rightarrow R^n$). Now consider a set of 2D features resulting from the projection of the 3D geometric primitives in the scene onto the image frame. Denoting $s = (s_1, s_2, \ldots, s_n)^T$ as the vector of parameters that characterizes these 2D features.

A scene feature is a set of tridimensional geometrical primitives (points, lines, vertices, etc.) rigidly linked to a single body. Without loss of generality, the focal length is assumed to be equal to 1 so that any point with coordinates $\overline{x} = (x, y, z)^T$ is projected on the image plane as a point with coordinates $\overline{X} = (X, Y, 1)^T$ with

$$\overline{X} = \frac{1}{z}\overline{x} \quad . \tag{1}$$

An image feature is a set of primitives in the image plane which corresponds to the projection of a scene feature. A configuration of the image feature is an element P_I.

Since our task here is to follow lines in 3D, we set our geometric feature parameters as lines, see figure 3-a. The canonical equation of a line in the image plane is $g(\overline{X}, \overline{P}) = AX + BY + C = 0$. However, this representation is not minimal (3 parameters (A,B,C) and $dim(P_I) = 2$). Using the representations $Y = -(\frac{A}{B}X + \frac{C}{B})$, with $B \neq 0$, or $X = -(\frac{B}{A}Y + \frac{C}{A})$, with $A \neq 0$, lead to use two charts on P_I, but some problems may occur due to the discontinuity.

Therefore, it is better to define lines using other representation (θ, ρ), which is minimal:

$$g(\overline{X}, \overline{P}) = X\cos\theta + Y\sin\theta - \rho = 0 \tag{2}$$

where the θ and the ρ parameters are defined by:

$$\theta = \arctan(\tfrac{B}{A}), \quad \rho = -\tfrac{C}{\sqrt{A^2+B^2}} \tag{3}$$

The ambiguity of this representation, since the same line may be parameterized by $(\rho, \theta + 2k\pi)$ and $(-\rho, \theta + (2k+1)\pi)$, is overcome by defining a direction for the lines, i.e., fixing the sign of ρ and the multiple choices of θ by modulating to $[0, 2\pi]$ the error $(\theta - \theta_d)$, where the subscript d denotes the desired value [10].

To perform the task execution as desired (stated in section 2), we need three lines as our image features. Thus, our vector of parameters is $s = (\theta_1, \rho_1, \theta_2, \rho_2, \theta_3, \rho_3)^T$, which represents the parameters of the central and lateral lines in the image plane. The purpose of the vision task is to extract and compute these feature parameters at each frame during robot motion, and deliver them to the control system.

3.2 Vision Tasks Implementation

Detection of lines in an image has a long tradition in the computer vision literature. Several methods have already appeared and generally start with a detection of edges in the image. A Hough transform approach is often used. However, it requires attention to several aspects to take into account, such as the accumulator resolution, peaks detection threshold, among others to avoid the detection of peaks generated by noise. Also, the choice for these values depends strongly on experimentation.

The vision system is responsible for detecting the lines and its parameters through the use of any method and, taking two different points on each detected line, for calculating those parameters by the expressions given in (3). There are different methods to segment and track features in images, each with their own advantages and drawbacks.

Mathematical Morphology (MM) is a powerful tool for digital image processing. Opposite to the classical definition of an image as an amplitude function of its coordinates, MM treats the image as a set of pixels. A good review on the use of MM in image processing can be found in [5]. To accomplish the vision task, the feature segmentation and tracking is performed in real images using the sequence of morphological filters given below:

1. An *Opening Top-Hat* is considered here as a preprocessing stage to enhance fine details.

2. An *Adaptive Threshold* for a rough segmentation.

3. A *Subtraction* by a image with blobs bigger than a predefined value.

4. An *Area Opening* to remove blobs with an area smaller than a pre-defined value, since it contains little dots.

5. An *Inf-Reconstruction* using the previous image as the marker and the rough-segmented image as the conditioning one to capture the whole road.

6. An *Adaptive Threshold* to obtain the final image whose value of all pixels are between lower and upper bounds;

An example of the segmentation phase for a single frame is shown in the figure 3. Notice that, because of the simplicity of the synthetic image, one may reduce the number of the filters, keeping only the three last ones without loss of relevant information.

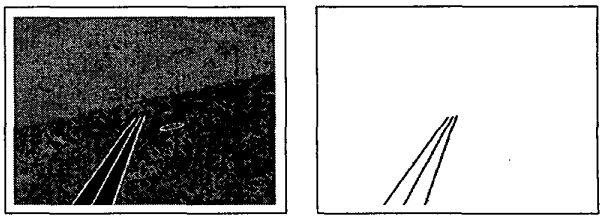

Figure 3: Image with addition of noise and the corresponding road segmentation.

To prove the validity of the filters suggested above, they were tested using real images and an example of the obtained results is depicted in the figure 4. The next step at each frame, after the segmentation phase, is to extract the vector of parameters s needed by the control stage.

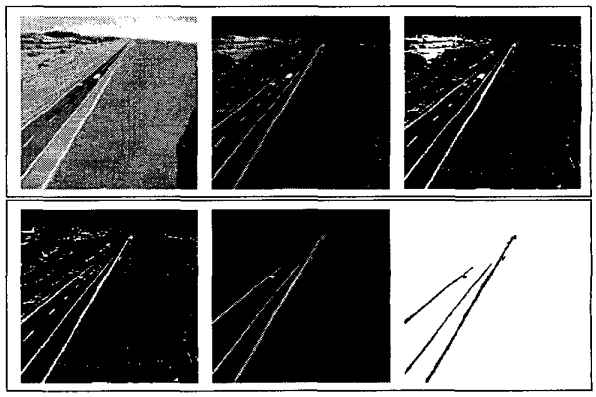

Figure 4: Sequence of transformations used to find the three lines from a real image of a highway. From up to down and left to right, it is shown the steps of the MM-based filtering and the lines obtained.

Although the algorithms for feature segmentation and tracking were implemented using mathematical morphology operators without any scheme of prediction or estimation (e.g., Kalman filters), the overall system performance is acceptable.

The authors are currently working on a hybrid strategy that combines the segmentation of the whole frame and parts of them, namely windows. Its main purpose is to

reduce the computational effort and, as consequence, the time delay induced by the unnecessary processing of the entire frame. A window is defined as a region of interest that contains the visual signal(s) required by the task being performed. Hence, the subsequent segmentations are conducted only within those areas, which ensures the complete task execution with minimal cost of processing.

The placement of each window in the frame plays an important role in the navigation of the robot. With the implementation of such strategy, the road following approach presented here is extended in the sense of the target being tracked. In this case, the visual tracking method may accept any kind of smooth curve, not only straight lines as visual signal (the set of frames still must provide all the components of the vector defined in section 3.1). The global tracking will be determined by the distance between those frames, i.e., this distance will define the horizon of prediction. Indeed, the closer the frames are, the closer the robot will follow the curve. □

4 Implementation of Control Aspects

Considering that vision aspects in the camera frame are solved, one has to derive the relationship between the 2D movement in the image frame to all 3D frames attached to the robot. These relationship may be viewed as Jacobians and, to be useful, they must consider all lack of informations due the image projection. Many authors discuss the control aspects of mobile robots; see for instance [13], [8], [11], [2] and the references therein for a good review on mobile robot modeling and control.

The approach proposed by Espiau *et al.* [3] basically consists of deriving a suitable mathematical relationship (the *interaction matrix* L^T) between the motion in the image plane of the parameters vector $\dot{s}$ and the motion of the 3D frame $\mathcal{F}_c$ attached to the camera T_c. T_c is composed by translational and rotational velocity, $T_c = [{}^c\dot{x}_c, {}^c\dot{y}_c, {}^c\dot{z}_c, {}^cw_x, {}^cw_y, {}^cw_z]^T$.

$$\dot{s} = L^T \cdot T_c \tag{4}$$

Using the general task function formalism, introduced in [11], we can express the robot task as regulating the output function $e(r,t)$:

$$e(r,t) = L^{T^+}\Big\|_{s^*} (s(r,t) - s^*) \tag{5}$$

where $s(\cdot)$ is the current visual features parameters related with the robot configuration r, $s^*(\cdot)$ are the desired values for $s(\cdot)$ and $L^{T^+}\Big\|_{s^*}$ is the pseudo-inverse of the interaction matrix, whose value computed at s^* is given by:

$$L^T\|_{s^*} = \begin{bmatrix} -\frac{\cos(\theta_{1d})^2}{h_{ref}} & -\frac{\cos(\theta_{1d})*\sin(\theta_{1d})}{h_{ref}} & 0 \\ 0 & 0 & 0 \\ -\frac{\cos(\theta_{2d})^2}{h_{ref}} & -\frac{\cos(\theta_{2d})*\sin(\theta_{2d})}{h_{ref}} & 0 \\ 0 & 0 & 0 \\ -\frac{\cos(\theta_{3d})^2}{h_{ref}} & -\frac{\cos(\theta_{3d})*\sin(\theta_{3d})}{h_{ref}} & 0 \\ 0 & 0 & 0 \end{bmatrix} \cdots \tag{6}$$

$$\cdots \begin{bmatrix} 0 & 0 & -1 \\ \sin(\theta_{1d}) & -\cos(\theta_{1d}) & 0 \\ 0 & 0 & -1 \\ \sin(\theta_{2d}) & -\cos(\theta_{2d}) & 0 \\ 0 & 0 & -1 \\ \sin(\theta_{3d}) & -\cos(\theta_{3d}) & 0 \end{bmatrix}$$

where h_{ref} is the desired height, θ_{1d}, θ_{2d} and θ_{3d} are the desired angles for the central e lateral lines.

Given the velocities of the 3D camera frame, one has to derive a non-singular Jacobian to express these velocities in terms of robot actuator speeds, and then, design a suitable controller. Both Jacobian and controller must consider spatial and/or mechanical constraints of the robotic platform.

As the objective of this paper is to solve the vision aspect, the detailed solution of the airship modeling and controller design issues is treated in [12]. To validate the vision solution, we considered a simplified model of the AUV.

An interesting aspect of this approach is the concept of virtual linkage between the sensor and the objects of the environment, which is based on the range of the interaction matrix $\mathcal{R}(L^T)$ and gives all motions of the camera that keep the image unchanged. In this paper we are interested in an interaction matrix that permits a *prismatic-like* motion, which can be implemented by utilizing three lines as a reference image for the road following task. For the remaining DOF, it is given a constant reference velocity ν for simplicity. Other trajectory tracking technique may also be applied.

The control law that will bring $e(r,t)$ to zero is selected to be as [3], computed in terms of $\mathcal{F}_c$ speed and for the DOFs constrained by the linkage.

$$T_c = -\lambda e \qquad \text{with} \quad \lambda > 0 \tag{7}$$

where λ constitutes the gain conditions for the airship actuators. Since the linearized non-linear model can be decoupled between lateral and longitudinal motions, one has to define separately those gains. Thus, the overall convergence will depend on this setup as well as the cruising speed. The results shown below clarify the behavior for different conditions, including an ideal, a real and a too slow convergence for an aerial robot like the airship taken here.

5 Results

The setup used to obtain the results presented here is based on the airship 6 DOF model depicted in figure 1. The simulation hardware was a workstation Sun ULTRA 10 (UltraSPARC IIi-330MHz) with 128Mb of RAM, where all the computation is monoprocessed and sequential and with little effort to permit the complete task execution.

The visual servoing scheme is composed by two independent processes: the Airship 6 DOF-model SIMULINK-based control system development environment and the visualizator, written in openGL, which is responsible to feedback the visual input. These processes communicate to each other through TCP/IP sockets and they are better understood by the figure 5.

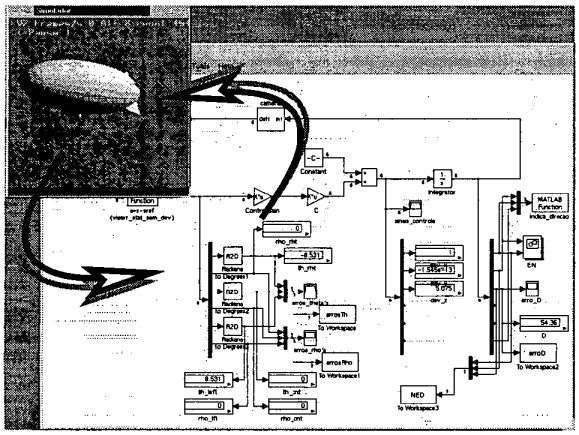

Figure 5: How the interconnection between the simulator and the visualizator was made.

Figure 6 shows the top-view behavior of the aerial robot during the task execution. Additionally, we have plotted the navigation with other headings and gain conditions to show that the feasibility of the system is quite independent of the starting point (the lines must be in the field of view). Notice that for the first tracking curve, the robot approximates the road very slowly and this may be undesirable. For the second one, we have a fast convergence, however this is not a typical behavior for the airship AS-800 since it has non-holonomic constraints and limits on its thrust propellers. The third curve is more realistic since it accounts for the physical properties of the robot.

Further investigation was carried on the third curve to illustrate the effectiveness of the proposed methodology. The behavior of the difference between the vector of parameters s computed at frame rate and its desired value s^*, $(s(r,t) - s^*)$, is shown below in the figure 7. They correspond to the error in the image plane as previously defined and capture the exponential convergence of the error.

The control signals for driving the robot actuators, and

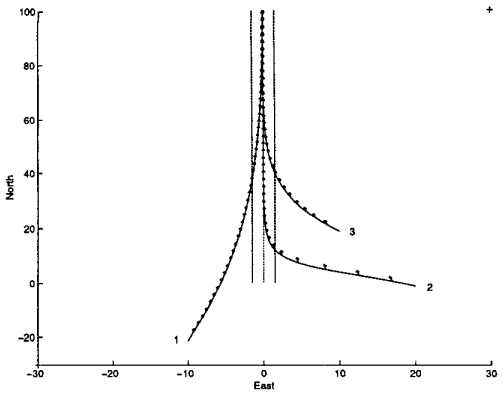

Figure 6: Bird's eye view of the road following behavior for different headings and gain conditions.

thus, the error signals depicted in figure 7 to zero are drawn in figure 8. The constant reference velocity ν was set to $8m/s$ and the gain λ to 0.6.

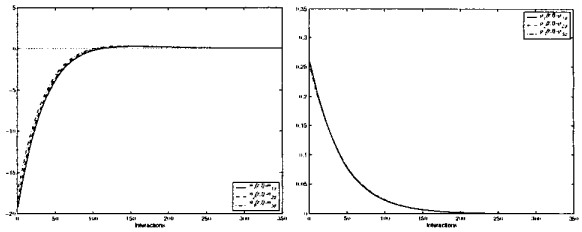

Figure 7: Evolution of the errors in the image plane for the visual features θ and ρ.

6 Conclusions

Applications of visual servo control is expected to be enormous in the future. Cameras are relatively inexpensive devices and the cost of image processing systems continues to fall.

In this paper, it is presented an application of a pure vision-based control. The task here, described in terms of image parameters, was to follow the road composed by three lines. After performing the vision tasks, which consist basically by the image segmentation, feature extraction and computation of feature parameters, the control aligns the aerial autonomous robot with those lines having an exponential convergence of the error.

Aerial Unmanned Vehicles, including airships, has received increasing interest over this past decade thanks to their enormous untapped potential in low-speed, low-altitude exploration, monitoring, biodiversity and climate research as well as telecommunication relay systems. In

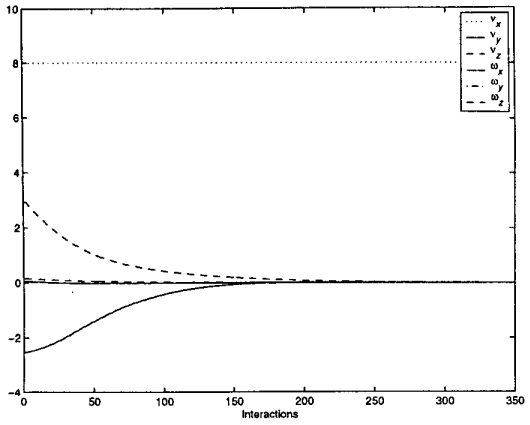

Figure 8: Control signals.

order to extend the autonomy level of this still partially explored platform, we developed and validated this new path tracking methodology through the results shown in the preceding sections.

A further step in this work will concentrate on implementing this system in the real aerial robot AS-800 produced by Airspeed Airships, which is in the context of the Project AURORA [1].

Acknowledgments

This project earns partial grant from FAPESP under no. 97/13384-7 and 98/13562-5 and CNPq/CTPETRO under grant no. 466713/00-2.

References

[1] AURORA: Autonomous Unmanned Remote Monitoring Robotic Airship. Home page. In *http://www.iti.br/projetos/aurora/motivation.htm.*

[2] S. S. Bueno, A. Elfes, M. Bergerman, and J. G. Ramos. A semi-autonomous robotic airship for environment monitoring missions. In *IEEE International Conference on Robotics and Automation*, Leuven, Belgium, May 1998.

[3] Bernard Espiau, Franois Chaumette, and Patrick Rives. A new approach to visual servoing in robotics. *IEEE Transactions on Robotics and Automation*, 8:313–326, 1992.

[4] Seth Hutchinson, Gregory D. Hager, and Peter I. Coke. A tutorial on visual servo control. *IEEE Transactions on Robotics and Automation*, 12(5):651–670, October 1996.

[5] Barrera J, Banon J F, and Lotufo R A. Image algebra and morphological image processing V. In *International Symposium on Optics, Imaging and Instrumentation, SPIE's Annual Meeting*, San Diego, USA, July 1994.

[6] Yi Ma, Jana Kosecka, and Shankar S. Sastry. Vision guided navigation for a nonholonomic mobile robot. *IEEE Transactions on Robotics and Automation*, 15(3):521–537, June 1999.

[7] K. Munson, ed. Jane's unmanned aerial vehicles and targets. *Jane's Information Group Limited*, 1996.

[8] Patrick Fred Muir. *Modelling an control of wheeled mobile robots.* PhD thesis, DECE - Carnegie Mellon University, USA, 1988.

[9] P. Rives and J.J Borrelly. Visual servoing techniques applied to an underwater vehicle. In *Proceedings of the 1997 International Conference on Robotics ans Automation*, Albuquerque, USA, april 1997.

[10] Patrick Rives. Lecture notes on sensor based control. March 1999.

[11] Claude Samson, Bernard Espiau, and M. le Borges. *Robot Control: the Task Function Approach.* Oxford University Press, USA, 1990.

[12] Geraldo F. Silveira, J. Reginaldo H. Carvalho, Marconi K. Madrid, Samuel S. Bueno, and Patrick Rives. Towards vision guided navigation of autonomous aerial robots. In *Proceedings of the IV Brazilian Symposium on Intelligent Automation*, Gramado/RS, Brazil, December 2001. submitted.

[13] Geraldo F. Silveira, J. Reginaldo H. Carvalho, Pedro Mitsuo Shiroma, Patrick Rives, and Samuel Siqueira Bueno. Visual servo control of nonholonomic mobile robots. In *16th Brazilian Congress of Mechanical Engineering*, Uberlandia, Brazil, November 2001. accepted.