Robot Servoing"
International Conference on Intelligent Robots
and Systems (IROS'97)
Grenoble, France, September 8-12, 1997

Applying Visual Servoing Techniques to Control Nonholonomic Mobile Robots

Dimitris P. Tsakiris, Patrick Rives and Claude Samson
INRIA Sophia-Antipolis
2004, Route des Lucioles, B.P. 93
06902, Sophia Antipolis Cedex - France
{first_name}.{last_name}@sophia.inria.fr

Abstract

The stabilization to a desired pose of a nonholonomic mobile robot, based on visual data from a hand-eye system mounted on it, is considered. Instances of this problem occur in practice during docking or parallel parking maneuvers of such vehicles. In this paper, we investigate the use of visual servoing techniques for their control. After briefly presenting the relevant visual servoing framework, we point out some problems encountered when it is considered for nonholonomic mobile robots. In particular, simple velocity control schemes using visual data as feedback cannot be applied anymore. We show how, by using the extra degrees of freedom provided by the hand-eye system, we can design controllers capable of accomplishing the desired task. A first approach, allows to perform a visual servoing task defined in the camera frame without explicitly controlling the pose of the nonholonomic mobile basis. A second approach based on continuous time–varying state feedback techniques allows to stabilize both the pose of the nonholonomic vehicle and that of the camera.

The experimental evaluation of the proposed techniques uses a mobile manipulator prototype developed in our laboratory and dedicated multiprocessor real–time image processing and control systems.

1 Introduction

In order to perform a task with a mobile robot, one needs to efficiently solve many interesting problems from task planning to control law synthesis. On the one hand, at the high level, problems of path planning and off–line programming have been successfully addressed. Efficient planners exist, using recent computational geometry results, which allow to compute free trajectories in the configuration space taking into account geometric and kinematic constraints [1]. On the other hand, at the control level, important results have been established for nonholonomic systems, like wheeled mobile robots, which lead to specific control problems: not only the linearization of these systems is uncontrollable, thus linear analysis and design methods cannot be applied in some interesting problems, but also there do not exist continuous feedback control laws, involving only the state, capable to stabilize such a system to an equilibrium, due to a topological obstruction pointed out by Brockett [2]. One of the approaches developed to solve the stabilization problem is the use of time–varying state feedback, i.e. control laws that depend explicitly, not only on the state, but also on time, usually in a periodic way, which Samson [18] introduced in the context of the unicycle's point stabilization. This sparked a significant research effort ([3], [5], [13], [10]), which demonstrated the existence of efficient such feedback control laws and provided some design procedures.

These results can be very useful in sensor–based control of mobile robotic systems. One of the prominent methods in this area is visual servoing, which was originally developed for manipulator arms with vision sensors mounted at their end–effector [6], [4], [7], [8]. In this paper, we point out the difficulties of transferring directly these techniques to nonholonomic mobile robots. We show, however, that by properly adding degrees–of–freedom to the nonholonomic platform, in the form of a hand–eye system, and by taking advantage of the time–varying stabilizing control schemes, it is still possible to extend visual servoing techniques to nonholonomic systems [12], [19], [20], [21]. For simplicity we only consider here the planar case, where a mobile robot of the unicycle type carries an n-d.o.f. planar manipulator arm with a camera that moves parallel to the plane supporting the mobile robot.

In section 2 of the paper we recall some notions from geometric mechanics relevant to the study of kinematics on the plane; we also provide an overview of the visual servoing framework, which allows to design robust vision–based closed–loop controls for holonomic robotic systems. In section 3, we model the kinematics and vision system of a nonholonomic mobile manipulator with an n–degree–of–freedom planar arm. Section 4 is dedicated to the analysis and synthesis of various

visual servoing control schemes for our system. The last section presents related experimental results.

2 Preliminaries

2.1 Kinematics on the Plane and the $SE(2)$ Group

The analysis of kinematic chains, like the ones present in robotic systems, is made significantly more compact by adopting the geometric viewpoint in mechanics, as detailed in e.g. [11]. A synopsis of relevant notions for our study of planar multi–d.o.f. kinematic chains is given below.

The relative configuration of any two coordinate systems on the plane can be described by an element g of the Special Euclidean group $SE(2)$ of rigid planar motions. A curve $g(.) \subset SE(2)$ can describe the relative displacement of the two coordinate systems during a motion of the system. The velocity at every point of this curve is an element of the tangent space of $SE(2)$, which is isomorphic to the Lie algebra $se(2)$ of this group. For $g \in SE(2)$ and $\xi \in se(2)$, we define the *adjoint action* of $SE(2)$ on its Lie algebra $se(2)$ as

$$Ad : SE(2) \times se(2) \longrightarrow se(2) : (g, \xi) \longmapsto Ad_g \xi = g\ \xi\ g^{-1}. \tag{1}$$

The isomorphism of velocities with elements of the Lie algebra can be defined in two ways, corresponding to spatial and body velocities, by left and right translations of the tangent space elements to the group identity. Then, differentiation of $g(t) \in SE(2)$ leads to

$$\dot{g} = \Xi\ g = g\ \xi, \tag{2}$$

where $\Xi, \xi \in se(2)$, are, respectively, the spatial and body velocity of the system's motion and which are related by

$$\Xi = Ad_g \xi \quad \text{and} \quad \xi = Ad_{g^{-1}} \Xi. \tag{3}$$

The corresponding coordinate–based relations are based on the homogeneous matrix representation of an element $g_{i,j} \in SE(2)$ (describing the relative configuration of coordinate system $\{F_j\}$ with respect to $\{F_i\}$) as

$$g_{i,j} = g_{i,j}(x_{i,j}, y_{i,j}, \theta_{i,j}) = \begin{pmatrix} \cos\theta_{i,j} & -\sin\theta_{i,j} & x_{i,j} \\ \sin\theta_{i,j} & \cos\theta_{i,j} & y_{i,j} \\ 0 & 0 & 1 \end{pmatrix}, \tag{4}$$

where $(x_{i,j}, y_{i,j})$ can be interpreted as the position of the center of the coordinate system $\{F_j\}$ with respect to that of $\{F_i\}$ and $\theta_{i,j}$ as the angle that the x–axis of the coordinate system $\{F_j\}$ makes with respect to that of $\{F_i\}$. The usual basis of $se(2)$ is $\{\mathcal{A}_1, \mathcal{A}_2, \mathcal{A}_3\}$, where these matrices are

$$\left\{ \begin{pmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}, \begin{pmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}, \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{pmatrix} \right\} \tag{5}$$

A generic element $\xi \in se(2)$ can be represented in this basis as

$$\xi = \omega \mathcal{A}_1 + \xi_1 \mathcal{A}_2 + \xi_2 \mathcal{A}_3 = \begin{pmatrix} 0 & -\omega & \xi_1 \\ \omega & 0 & \xi_2 \\ 0 & 0 & 0 \end{pmatrix}, \tag{6}$$

with $\omega, \xi_1, \xi_2 \in I\!R$.

Let $\Xi_{i,j},\ \xi_{i,j} \in se(2)$ be the velocities corresponding (as specified by equation 2) to $g_{i,j}$ and let the coordinates of $\Xi_{i,j}$ be $(\Omega_{i,j},\ \Xi_1^{i,j},\ \Xi_2^{i,j})$ and those of $\xi_{i,j}$ be $(\omega_{i,j},\ \xi_1^{i,j},\ \xi_2^{i,j})$. Then, the equations 2 take the form:

$$\begin{aligned} \dot{\theta}_{i,j} &= \Omega_{i,j} = \omega_{i,j}, \\ \dot{x}_{i,j} &= \Xi_1^{i,j} - \Omega_{i,j}\ y_{i,j} = \xi_1^{i,j} \cos\theta_{i,j} - \xi_2^{i,j} \sin\theta_{i,j}, \\ \dot{y}_{i,j} &= \Xi_2^{i,j} + \Omega_{i,j}\ x_{i,j} = \xi_1^{i,j} \sin\theta_{i,j} + \xi_2^{i,j} \cos\theta_{i,j}, \end{aligned} \tag{7}$$

The above notation will be used throughout our kinematic calculations.

2.2 Visual Servoing

We consider a special case of the general visual servoing framework developed in [6], [4] and surveyed in [7], [8], [16], as it applies to a hand–eye system composed of a manipulator arm with a camera mounted on its end–effector. We want to drive the camera to a desired pose (position and orientation) with respect to a target of interest. Let X be the system state (e.g. the manipulator arm joint coordinates q or the camera Cartesian coordinates X_C) and Y be the camera data (e.g. the target features). From the manipulator arm kinematics, the camera projection model and the scene geometry, we can derive the relationship between X and Y : $Y = \Phi(X)$, as well as the corresponding differential relationship

$$\dot{Y} = \frac{\partial \Phi}{\partial X}(X)\ \dot{X}\ . \tag{8}$$

If X is the camera Cartesian coordinates X_C and $\dot{X}$ is the spatial velocity of the target with respect to the camera, then $\frac{\partial \Phi}{\partial X}$ corresponds to the *interaction matrix* $L^\top$ of [6]. If X is the manipulator arm joint coordinates q, then only one part of $\frac{\partial \Phi}{\partial X}$ corresponds to the interaction matrix, while the other is given by the manipulator Jacobian.

A robot task, e.g. that of driving the camera to a desired pose $X^\star$, can be expressed in the *task function framework* of [17] as the regulation of an appropriately chosen task function e to zero. Let the sensory data at the desired pose be $Y^\star$. Assuming that the target is

static, we can choose as a task function the deviation of the sensory data from the desired ones:

$$e(X) = \Phi(X) - Y^\star \ . \tag{9}$$

To regulate this task function e exponentially to 0, we can choose $\dot{X}$, which we can consider as our control, so that the corresponding closed–loop system is of the form

$$\dot{e} = -\lambda \ e, \tag{10}$$

for a positive gain λ. Then, if the state X is not too far away from the desired one, the control $\dot{X}$ is specified by equations 8, 9 and 10 as

$$\dot{X} = -\lambda \left[\frac{\partial e}{\partial X}(X)\right]^\dagger (Y - Y^\star) \tag{11}$$

where $\left[\frac{\partial e}{\partial X}(X)\right]^\dagger$ is the pseudo–inverse of $\frac{\partial e}{\partial X}$ (in case $\dim(X) = \dim(e)$, this is simply the inverse of $\frac{\partial e}{\partial X}$). A frequently–used simplification is to evaluate $\left[\frac{\partial e}{\partial X}\right]^\dagger$ in equation 11 only at the desired final state $X_\star$.

If X is the manipulator arm joint coordinates q, this gives directly a velocity control scheme to achieve the desired task. The $\dot{X}$ specified in this control law will be implemented via the manipulator arm and the desired task will be executed.

In the case that nonholonomic constraints on the motion of the system are present, however, as is the case when the manipulator arm is mounted on a mobile robot, the above approach of sensor–based control does not translate directly to the resulting system. If X is, now, the state of this system, there exist velocities $\dot{X}$ that cannot be implemented due to the constraints. Several possible ways to bypass these restrictions are described below.

3 Mobile Manipulator Modeling

3.1 Mobile Manipulator Kinematics

We consider a mobile robot of the unicycle type carrying an n-d.o.f. planar manipulator arm with a camera mounted on its end effector (figure 1 shows the case of $n = 3$).

Consider an inertial coordinate system $\{F_O\}$ centered at a point O of the plane, a moving coordinate system $\{F_M\}$ attached to the middle M of the robot's wheel axis and another moving one $\{F_C\}$ attached to the optical center C of the camera. Let (x, y) be the position of the point M and θ be the orientation of the mobile robot with respect to the coordinate system $\{F_O\}$; let l_m be the distance of the point M from the first joint B_1 of the n-d.o.f. planar arm, with $l_1, \ \ldots \ , l_n$ being the lengths of the links of the arm and $\psi_1, \ \ldots \ , \psi_n$ being its joint coordinates.

Let $g_{_M} = g_{_M}(x, y, \theta)$ represent the configuration of the coordinate system $\{F_M\}$ with respect to $\{F_O\}$,

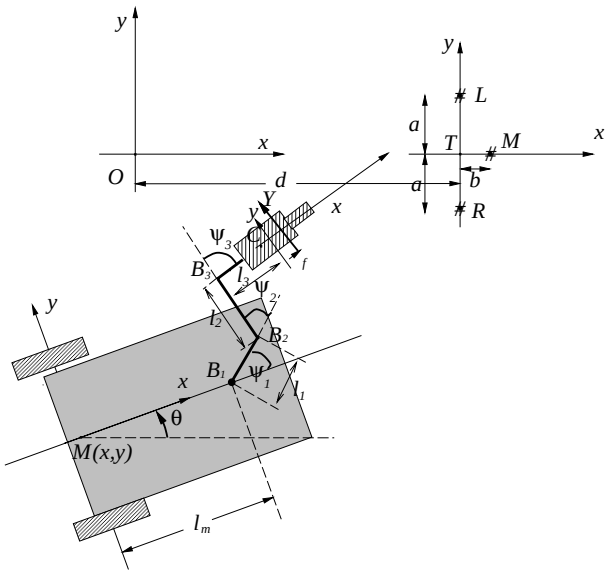

Figure 1: Mobile Manipulator with Camera

let $g_{_{MC}} = g_{_{MC}}(x_{MC}, y_{MC}, \theta_{MC})$ represent the configuration of $\{F_C\}$ with respect to $\{F_M\}$, let $g_{_{CT}} = g_{_{CT}}(x_{CT}, y_{CT}, \theta_{CT})$ represent the configuration of $\{F_T\}$ with respect to $\{F_C\}$, let $g_{_C} = g_{_C}(x_C, y_C, \theta_C)$ represent the configuration of $\{F_C\}$ with respect to $\{F_O\}$ and let $g_{_T} = g_{_T}(x_T, y_T, \theta_T)$ represent the configuration of $\{F_T\}$ with respect to $\{F_O\}$, where $x_T = d$ is the distance of point T from point O and $y_T = \theta_T = 0$.

From the kinematic chain of figure 1 we have for the case of an n–degree–of–freedom manipulator arm:

$$g_{_{MC}} = g_{_{MB_1}} g_{_{B_1B_2}} \cdots g_{_{B_nC}}, \tag{12}$$

where

$$g_{_{MB_1}} = \begin{pmatrix} 1 & 0 & l_m \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} , \tag{13}$$

$$g_{_{B_iB_{i+1}}} = \begin{pmatrix} \cos\psi_i & -\sin\psi_i & l_i\cos\psi_i \\ \sin\psi_i & \cos\psi_i & l_i\sin\psi_i \\ 0 & 0 & 1 \end{pmatrix} , \tag{14}$$

for $i = 1, ..., n$ and where we identify the point B_{n+1} with C. It is easy to see that

$$\begin{aligned} \theta_{MC} &= \sum_{i=1}^{n} \psi_i \ , \\ x_{MC} &= l_m + \sum_{i=1}^{n} l_i \cos\big(\sum_{j=1}^{i} \psi_j\big) \ , \\ y_{MC} &= \sum_{i=1}^{n} l_i \sin\big(\sum_{j=1}^{i} \psi_j\big) \ . \end{aligned} \tag{15}$$

From $g_{C} = g_{M} g_{MC}$ we get

$$\begin{aligned} \theta_C &= \theta + \sum_{i=1}^{n} \psi_i \ , \\ x_C &= x + l_m \cos\theta + \sum_{i=1}^{n} l_i \cos\big(\theta + \sum_{j=1}^{i} \psi_j\big) \ , \\ y_C &= y + l_m \sin\theta + \sum_{i=1}^{n} l_i \sin\big(\theta + \sum_{j=1}^{i} \psi_j\big) \ . \end{aligned} \tag{16}$$

From the kinematic chain of figure 1 we have

$$g_{T} = g_{M}\ g_{MC}\ g_{CT} \ . \tag{17}$$

From this, we obtain

$$g_{CT} = g_{MC}^{-1} g_{M}^{-1} g_{T} \tag{18}$$

and

$$g_{M} = g_{T} g_{CT}^{-1} g_{MC}^{-1} . \tag{19}$$

The corresponding coordinate relationships are

$$\begin{aligned} \theta_{CT} &= \theta_T - \theta_C \\ x_{CT} &= -(x_C - x_T)\cos\theta_C - (y_C - y_T)\sin\theta_C \ , \\ y_{CT} &= (x_C - x_T)\sin\theta_C - (y_C - y_T)\cos\theta_C \ , \end{aligned} \tag{20}$$

and

$$\begin{aligned} \theta &= \theta_T - \theta_{MC} - \theta_{CT} \ , \\ x &= x_T - x_{CT}\cos(\theta_T - \theta_{CT}) + y_{CT}\sin(\theta_T - \theta_{CT}) \\ &\quad - x_{MC}\cos\theta + y_{MC}\sin\theta \ , \\ y &= y_T - x_{CT}\sin(\theta_T - \theta_{CT}) - y_{CT}\cos(\theta_T - \theta_{CT}) \\ &\quad - x_{MC}\sin\theta - y_{MC}\cos\theta \ . \end{aligned} \tag{21}$$

Equations 18 and 20 are useful in simulating our system, while 19 and 21 are useful in reconstructing its state, supposing we know g_{CT}.

Velocity Kinematics: By differentiating the chain kinematics of the mobile robot and arm system (equation 12), we can specify the spatial velocity Ξ_{MC} as:

$$\Xi_{MC} = \sum_{i=1}^{n} Ad_{g_{MB_i}} \Xi_{B_i B_{i+1}} \ , \tag{22}$$

where $\Xi_{B_i B_{i+1}} = \dot{\psi}_i\ \mathcal{A}_1$ and $\mathcal{A}_1$ is the matrix specified in 5. The corresponding coordinate relationships are

$$\begin{aligned} \omega_{MC} &= \sum_{i=1}^{n} \dot{\psi}_i \ , \\ \Xi_1^{MC} &= \sum_{i=1}^{n} \dot{\psi}_i \sum_{j=1}^{i-1} l_j \sin\big(\sum_{k=1}^{j} \psi_k\big) \ , \\ \Xi_2^{MC} &= -\sum_{i=1}^{n} \dot{\psi}_i \Big[l_m + \sum_{j=1}^{i-1} l_j \cos\big(\sum_{k=1}^{j} \psi_k\big)\Big] \ . \end{aligned} \tag{23}$$

By differentiating the chain kinematics of the whole system and its environment (equation 17), assuming that we consider stationary targets ($\dot{g}_{T} = 0$) and solving for the spatial velocity Ξ_{CT}, we get

$$\Xi_{CT} = -Ad_{g_{MC}^{-1}}\big(\xi_M + \Xi_{MC}\big) \ , \tag{24}$$

where the body velocity ξ_M, which corresponds to g_{M}, is

$$\begin{pmatrix} 0 & -\dot{\theta} & \dot{x}\cos\theta + \dot{y}\sin\theta \\ \dot{\theta} & 0 & -\dot{x}\sin\theta + \dot{y}\cos\theta \\ 0 & 0 & 0 \end{pmatrix} . \tag{25}$$

Using 22, equation 24 becomes

$$\Xi_{CT} = -Ad_{g_{MC}^{-1}}\ \xi_M - \sum_{i=1}^{n} \dot{\psi}_i \big(Ad_{g_{B_iC}^{-1}}\ \mathcal{A}_1\big) \ . \tag{26}$$

This is a linear relationship between the coordinates of Ξ_{CT}, the coordinates of ξ_M and the $\dot{\psi}_i$'s, which leads to the associated Jacobian. The corresponding coordinate relationships are

$$\begin{aligned} \omega_{CT} &= -\dot{\theta} - \sum_{i=1}^{n} \dot{\psi}_i \ , \\ \Xi_1^{CT} &= -\dot{x}\cos\theta_C - \dot{y}\sin\theta_C \\ &\quad -\dot{\theta}\Big[l_m \sin\big(\sum_{i=1}^{n} \psi_i\big) + \sum_{i=1}^{n-1} l_i \sin\big(\sum_{j=i+1}^{n} \psi_j\big)\Big] \\ &\quad - \sum_{i=1}^{n-1} \dot{\psi}_i \sum_{j=i}^{n-1} l_j \sin\big(\sum_{k=j+1}^{n} \psi_k\big) \ , \\ \Xi_2^{CT} &= \dot{x}\sin\theta_C - \dot{y}\cos\theta_C \\ &\quad -\dot{\theta}\Big[l_m \cos\big(\sum_{i=1}^{n} \psi_i\big) + \sum_{i=1}^{n-1} l_i \cos\big(\sum_{j=i+1}^{n} \psi_j\big) + l_n\Big] \\ &\quad - \sum_{i=1}^{n-1} \dot{\psi}_i \Big[\sum_{j=i}^{n-1} l_j \cos\big(\sum_{k=j+1}^{n} \psi_k\big) + l_n\Big] - \dot{\psi}_n\ l_n \ . \end{aligned} \tag{27}$$

From this, the corresponding Jacobian can be read out. It is the $3 \times (n+3)$ matrix $(B_{1,1}\ B_{1,2})$ below

$$\Xi^{CT} \stackrel{\text{def}}{=} \begin{pmatrix} \Xi_1^{CT} \\ \Xi_2^{CT} \\ \omega_{CT} \end{pmatrix} = \begin{pmatrix} B_{1,1} & B_{1,2} \end{pmatrix} \begin{pmatrix} \dot{\mathcal{X}} \\ \dot{q} \end{pmatrix} \ , \tag{28}$$

where $\mathcal{X} \stackrel{\text{def}}{=} (x, y, \theta)^{\top}$ is the state of the mobile robot, while $q \stackrel{\text{def}}{=} (\psi_1, \psi_2, ..., \psi_n)^{\top}$ is the configuration of the manipulator arm and where the matrix $B_{1,1}$ is

$$\begin{pmatrix} -\cos\theta_C & -\sin\theta_C & b_1^{1,1} \\ \sin\theta_C & -\cos\theta_C & b_2^{1,1} \\ 0 & 0 & -1 \end{pmatrix} , \tag{29}$$

with θ_C given in 16, $b_1^{1,1} \stackrel{\text{def}}{=} -l_m \sin\left(\sum_{i=1}^{n} \psi_i\right) - \sum_{i=1}^{n-1} l_i \sin\left(\sum_{j=i+1}^{n} \psi_j\right)$, $b_2^{1,1} \stackrel{\text{def}}{=} -l_m \cos\left(\sum_{i=1}^{n} \psi_i\right) - \sum_{i=1}^{n-1} l_i \cos\left(\sum_{j=i+1}^{n} \psi_j\right) - l_n$ and the $3 \times n$ matrix $B_{1,2}$, which is the Jacobian of the manipulator arm, is

$$\begin{pmatrix} -\sum_{i=1}^{n-1} l_i \sin\left(\sum_{j=i+1}^{n} \psi_j\right) & \cdots & \cdots & 0 \\ -\left[\sum_{i=1}^{n-1} l_i \cos\left(\sum_{j=i+1}^{n} \psi_j\right) + l_n\right] & \cdots & \cdots & -l_n \\ -1 & \cdots & \cdots & -1 \end{pmatrix} . \quad (30)$$

Nonholonomic Constraints: The nonholonomic constraints on the motion of the mobile robot arise from the rolling–without–slipping of the mobile platform's wheels on the plane supporting the system and constrain its instantaneous motion, whose component lateral to the heading direction has to be zero, i.e.

$$\xi_2^M \stackrel{\text{def}}{=} -\dot{x} \sin\theta + \dot{y} \cos\theta = 0 . \quad (31)$$

From this and equations 2, we get the usual unicycle kinematic model for the mobile robot:

$$\dot{x} = v \cos\theta , \quad \dot{y} = v \sin\theta , \quad \dot{\theta} = \omega , \quad (32)$$

where $v \stackrel{\text{def}}{=} \dot{x} \cos\theta + \dot{y} \sin\theta$ is the heading speed and ω is the angular velocity of the unicycle. Then

$$\dot{\mathcal{X}} = B_{3,1}(X) \begin{pmatrix} v \\ \omega \end{pmatrix} , \quad (33)$$

where the matrix $B_{3,1}$ is

$$\begin{pmatrix} \cos\theta & 0 \\ \sin\theta & 0 \\ 0 & 1 \end{pmatrix} . \quad (34)$$

3.2 Vision Model

Consider a fixed target containing three easily identifiable feature points arranged in the configuration of figure 1 and let $\{F_T\}$ be the related coordinate frame, which we suppose to be translated along the x–axis of $\{F_O\}$ by a distance d. The coordinates of the three feature points with respect to $\{F_T\}$ are $(x_p^{\{T\}}, y_p^{\{T\}})$, $p \in \{l, m, r\}$. The distances a and b (fig. 1) are assumed to be known. The coordinates of the feature points with respect to $\{F_C\}$ are:

$$\begin{array}{rcl} x_p^{\{C\}} & = & x_{CT} + x_p^{\{T\}} \cos\theta_{CT} - y_p^{\{T\}} \sin\theta_{CT}, \\ y_p^{\{C\}} & = & y_{CT} + x_p^{\{T\}} \sin\theta_{CT} + y_p^{\{T\}} \cos\theta_{CT}. \end{array} \quad (35)$$

We consider the usual pinhole camera model for our vision sensor, with perspective projection of the target's feature points (viewed as points on the plane $\mathbb{R}^2$) on a 1–dimensional image plane (analogous to a linear CCD array). This defines the projection function $\mathcal{P}$ of a point of $\mathbb{R}^2$, which has coordinates (x, y) with respect to the camera coordinate frame $\{C\}$, as

$$\mathcal{P} : \mathbb{R}_+ \times \mathbb{R} \longrightarrow \mathbb{R} : (x, y) \longmapsto \mathcal{P}(x, y) = f \frac{y}{x}. \quad (36)$$

where f is the focal length of the camera. In our setup, the coordinate x corresponds to "depth".

Let the projections of the target feature points on the image plane be $Y_p = \mathcal{P}\left(x_p^{\{C\}}, y_p^{\{C\}}\right)$, $p \in \{l, m, r\}$, given by 36 and 35. The vision data are then $Y_v \stackrel{\text{def}}{=} (Y_l, Y_m, Y_r)^\top$. Differentiating 36, we get the well–known equations of the optical flow [9] for the 1–dimensional case:

$$\dot{Y}_v = B_{2,1}\left(Y_p, x_p^{\{C\}}\right) \Xi^{CT} , \quad (37)$$

where the matrix $B_{2,1}\left(Y_p, x_p^{\{C\}}\right)$ is:

$$\begin{pmatrix} -\frac{1}{x_l^{\{C\}}} Y_l & \frac{1}{x_l^{\{C\}}} f & \frac{1}{f}(f^2 + Y_l^2) \\ -\frac{1}{x_m^{\{C\}}} Y_m & \frac{1}{x_m^{\{C\}}} f & \frac{1}{f}(f^2 + Y_m^2) \\ -\frac{1}{x_r^{\{C\}}} Y_r & \frac{1}{x_r^{\{C\}}} f & \frac{1}{f}(f^2 + Y_r^2) \end{pmatrix} \quad (38)$$

and it corresponds to the interaction matrix L^T of [6].

3.3 The Full System

From the above modeling of the mobile robot with the n–d.o.f. manipulator arm, we get, in summary, the following results:

The state of the system is $X = (\mathcal{X}, q)^\top$. Then

$$\begin{pmatrix} \Xi^{CT} \\ \dot{q} \end{pmatrix} = B_1(X) \dot{X} , \quad (39)$$

where B_1 is the $(n+3) \times (n+3)$ matrix

$$\begin{pmatrix} B_{1,1} & B_{1,2} \\ \mathbf{0}_{n \times 3} & \mathbb{I}_{n \times n} \end{pmatrix} . \quad (40)$$

The sensory data are $Y = (Y_v, q)^\top$. Then

$$\dot{Y} = B_2(X) \begin{pmatrix} \Xi^{CT} \\ \dot{q} \end{pmatrix} \quad (41)$$

where B_2 is the $(n+3) \times (n+3)$ matrix

$$\begin{pmatrix} B_{2,1} & \mathbf{0}_{3 \times n} \\ \mathbf{0}_{n \times 3} & \mathbb{I}_{n \times n} \end{pmatrix} . \quad (42)$$

The relationship between the state and the sensory data $Y = \Phi(X)$ is given by equations 36, 35, 20 and 16. The corresponding differential relationship is

$$\dot{Y} = \frac{\partial \Phi}{\partial X}(X) \, \dot{X} = B_2(X) B_1(X) \, \dot{X}. \quad (43)$$

The controls of the system are $\mathcal{U} = (v,\ \omega,\ \omega_{\psi_1},\ \cdots, \omega_{\psi_n})^\top$. Then

$$\dot{X} = B_3(X)\,\mathcal{U}\ , \tag{44}$$

where B_3 is the $(n+3)\times(n+2)$ matrix

$$\begin{pmatrix} B_{3,1} & \mathbf{0}_{3\times n} \\ \mathbf{0}_{n\times 2} & I\!I_{n\times n} \end{pmatrix}\ . \tag{45}$$

4 Vision–based Control of Mobile Manipulators

4.1 Camera Pose Stabilization:

In this first approach, we show that it is possible to use a velocity control similar to the holonomic case, if the control objective does not require to explicitly control the pose of the mobile platform.

To illustrate that, we consider a reduced system with only one actuated pan-axis ($n = 1$). Our objective is to stabilize the camera to a desired pose, which is specified by the corresponding vision data $Y_v^\star \stackrel{\text{def}}{=} (Y_l^\star,\ Y_m^\star,\ Y_r^\star)^\top$. We select the task function $e(X) = Y_v - Y_v^\star$ with dim $e = 3$, which we want to drive exponentially to zero.

The system state is $X = (x, y, \theta, \psi_1)^\top$, the sensory data are $Y = (Y_l,\ Y_m\ Y_r,\ \psi_1)^\top$ and the control is $\mathcal{U} = (v, \omega, \omega_{\psi_1})^\top$. From 37 we get

$$\dot{e} = \dot{Y}_v = B_{2,1}(X)\,\Xi^{CT} = -\lambda\, e\ . \tag{46}$$

Then, away from singularities of $B_{2,1}$, we have

$$\Xi^{CT} = -\lambda\, B_{2,1}^{-1}(X)(Y_v - Y_v^\star)\ . \tag{47}$$

From the system kinematics we have

$$\Xi^{CT} = \big(B_{1,1}(X)\ B_{1,2}(X)\big) B_3(X)\,\mathcal{U}\ , \tag{48}$$

where the 3×4 matrix $\big(B_{1,1}\ B_{1,2}\big)$ is given by 29 and 30, and the 4×3 matrix B_3 is given by 45, by setting $n = 1$. The product $\big(B_{1,1}\ B_{1,2}\big)B_3$ is

$$\begin{pmatrix} -\cos\psi_1 & -l_m \sin\psi_1 & 0 \\ \sin\psi_1 & -[l_m\cos\psi_1 + l_1] & -l_1 \\ 0 & -1 & -1 \end{pmatrix} \tag{49}$$

and it is a nonsingular matrix, since its determinant is $-l_m$. Then

$$\mathcal{U} = \Big[\big(B_{1,1}(X)\ B_{1,2}(X)\big) B_3(X)\Big]^{-1}\ \Xi^{CT} \tag{50}$$

and finally from this and 47 we get

$$\mathcal{U} = -\lambda\Big[\big(B_{1,1}(X)\ B_{1,2}(X)\big) B_3(X)\Big]^{-1}\ B_{2,1}^{-1}(X^\star)(Y_v - Y_v^\star)\ . \tag{51}$$

Under this control law, the mobile manipulator moves from its initial configuration until the camera reaches its desired pose with respect to the target. However, the final pose of the mobile platform cannot be controlled at will and depends on its initial position and orientation. No controlled d.o.f.s are available in this case to correct this, while the task is executed.

Related experimental results obtained in [12], [15] are shown in fig. 2, where the trajectories of the system for two different initial configurations, but with the same desired camera pose with respect to the target, are plotted. The different final poses of the mobile platform can be seen. Fig. 3 shows the exponential convergence of the task function components (left), as well as the components of the velocity vector Ξ^{CT}, (right) during the experiments.

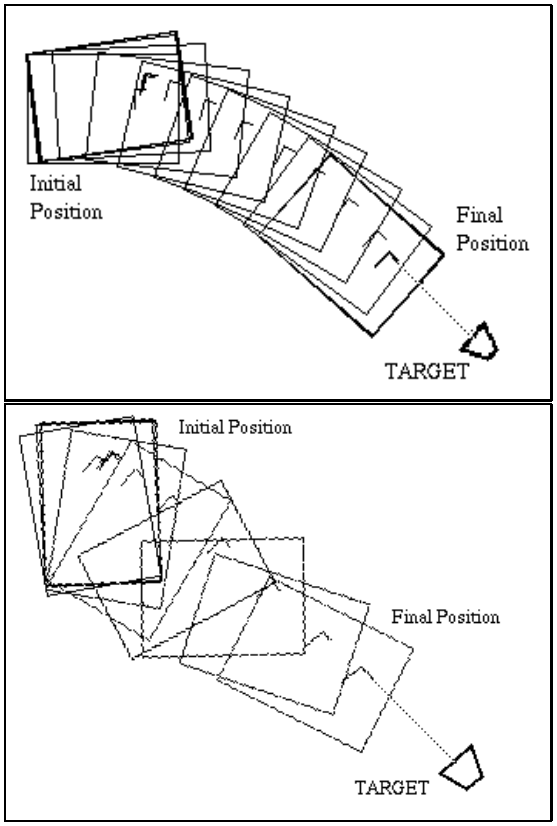

Figure 2: Robot trajectories for two different initial configurations and the same desired camera pose

4.2 Mobile Base Pose Stabilization

Consider the same system as in section 4.1 (i.e. the mobile robot with only one actuated pan-axis). Its state, sensory data and controls are also as before.

In this second approach, we consider the stabilization of the mobile platform to a desired pose with respect to some target. At the same time, we require

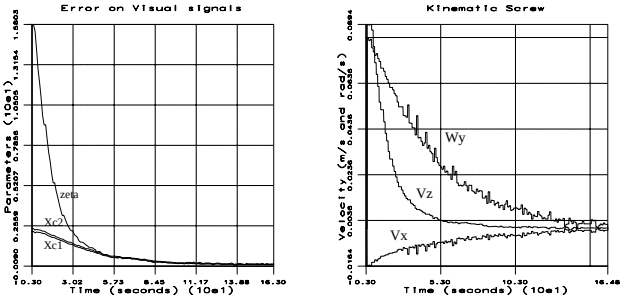

Figure 3: Evolution of visual data and velocity Ξ^{CT}

that the camera tracks the targets, despite the motion of the platform. The role of the arm is, in this case, to provide an extra d.o.f., which will allow the camera to move independently.

One of the approaches developed to solve the point stabilization problem for nonholonomic mobile robots is the use of time–varying state feedback, i.e. control laws that depend explicitly, not only on the state, but also on time, usually in a periodic way. Samson [18] introduced them in the context of the unicycle's point stabilization and raised the issue of the rate of convergence to the desired equilibrium. In this section, we apply techniques of time–varying state feedback, recently developed by Morin and Samson [10], in a visual servoing framework [19], [20].

The problem that we consider is to stabilize the mobile platform to the desired configuration which, without loss of generality, will be chosen to be zero, i.e. $X^\star = (x^\star, y^\star, \theta^\star, \psi_1^\star) = 0$. The corresponding visual data $Y_v^\star = (Y_l^\star, Y_m^\star, Y_r^\star)$ can be directly measured by putting the system in the desired configuration or can be easily specified, provided d is also known, along with the target geometry a and b (see figure 1).

An exponentially stabilizing control is considered for the mobile platform, while a control that keeps the targets foveated is considered for the camera.

Mobile platform control synthesis : In order to facilitate the synthesis of the controller, we apply a locally diffeomorphic transformation of the states and inputs

$$\begin{array}{rcl}(x_1, x_2, x_3)^\top & = & \Psi(X) \stackrel{\text{def}}{=} (x, y, \tan\theta)^\top \\ u_1 & = & \cos\theta\, v\ , \quad u_2 = \frac{1}{\cos^2\theta}\,\omega\ ,\end{array} \tag{52}$$

which brings the unicycle kinematics (equations 32) in the so-called *chained form* [13], [10]:

$$\dot{x}_1 = u_1\ ,\ \ \dot{x}_2 = x_3 u_1\ ,\ \ \dot{x}_3 = u_2\ . \tag{53}$$

The mobile platform control, that can be used if the state is known or reconstructed, is given by:

$$\begin{array}{rcl}v(t,X) & = & \frac{1}{\cos\theta}\, u_1(t, \Psi(X))\ , \\ \omega(t,X) & = & \cos^2\theta\, u_2(t, \Psi(X))\ ,\end{array} \tag{54}$$

where u_1 and u_2 are the time–varying state–feedback controls, developed by Morin and Samson [10] for the 3–dimensional 2–input chained–form system and which are given in terms of the chained–form coordinates of equation 52 by:

$$\begin{array}{rcl}u_1(t,x_1,x_2,x_3) & = & k_1\big[\rho_3(x_2,x_3) + \alpha(-x_1 \sin \mathrm{w}t \\ & & \qquad + |x_1 \sin \mathrm{w}t|)\big] \sin \mathrm{w}t\ , \\ u_2(t,x_1,x_2,x_3) & = & -\frac{k_3}{\rho_3(x_2,x_3)}\big[|u_1| x_3 + k_2 u_1 \frac{x_2}{\rho_2(x_2)}\big]\ ,\end{array} \tag{55}$$

where $\rho_2(x_2) \stackrel{\text{def}}{=} |x_2|^{\frac{1}{3}}$, $\rho_3(x_2,x_3) \stackrel{\text{def}}{=} \big(|x_2|^2 + |x_3|^3\big)^{\frac{1}{6}}$, w is the frequency of the time–varying controls and α, k_1, k_2, k_3 are positive gains.

The exponential convergence to zero of the closed–loop system can be demonstrated using the homogeneous norm $\rho(x_1,x_2,x_3) \stackrel{\text{def}}{=} \big(|x_1|^6 + |x_2|^2 + |x_3|^3\big)^{\frac{1}{6}}$.

The control $\mathcal{U}$ for the mobile platform is then

$$\mathcal{U}(t,X) = \big(v(t,X),\ \omega(t,X)\big)^\top. \tag{56}$$

Such a control requires an estimate $\hat{X}$ of the current state X. This estimate can be provided by state reconstruction from the visual data [19]. However, since we are interested in positioning the mobile robot to the desired configuration $X^\star = 0$, while starting relatively close to it, we could attempt to do so without reconstructing explicitly its state. Since $Y = \Phi(X)$, the state X can be approximated, near the configuration $X^\star = 0$, up to first order by

$$\hat{X}(Y) = \big[\frac{\partial \Phi}{\partial X}(X^\star)\big]^{-1}(Y - Y^\star)\ , \tag{57}$$

where $\frac{\partial \Phi}{\partial X} = B_2(X) B_1(X)$ with B_1 and B_2 as specified in 40 and 42 by setting $n = 1$. The proposed control law for the mobile platform can thus be expressed as a function of only the sensory data

$$\mathcal{U} = \mathcal{U}\big(t, Y\big). \tag{58}$$

Arm control synthesis : In order to implement a vision–based state–feedback control law for the mobile platform, we have to track the target during the motion of the platform. The arm control ω_{ψ_1} is chosen to keep the targets foveated by regulating the angular deviation of the line–of–sight of the camera from the targets to zero, while the mobile robot moves. It is specified so that Y_m is made to decrease exponentially to $Y_m^\star$, by regulating the task function $e(X) \stackrel{\text{def}}{=} Y_m - Y_m^\star$ to zero and by making the closed–loop system for e behave like $\dot{e} = -\lambda\, e$, for a positive gain λ. This gives

$$\omega_{\psi_1}(t,X,Y) = -\frac{\lambda}{\mathcal{J}_{2,3}}(Y_m - Y_{m\star}) - \left(\frac{\mathcal{J}_{2,1}}{\mathcal{J}_{2,3}}\, v + \frac{\mathcal{J}_{2,2}}{\mathcal{J}_{2,3}}\,\omega\right), \tag{59}$$

where $\mathcal{J}_{2,i}$ is the $(2,i)$–entry of the matrix $\mathcal{J}(X) \stackrel{\text{def}}{=} B_2(X)\ B_1(X)\ B_3(X)$. In particular, $\mathcal{J}_{2,3} = -f - \big(\frac{Y_m^2}{f} +$

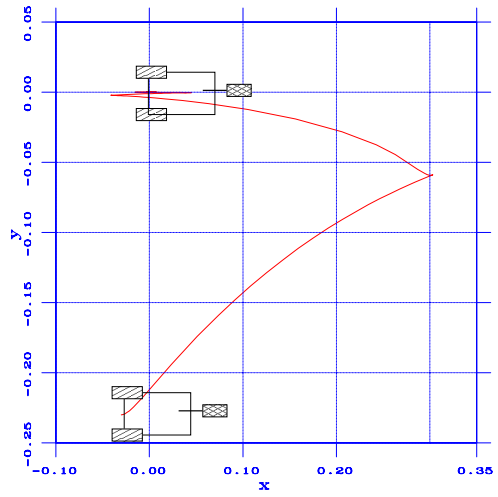

Figure 4: Mobile robot (x, y)–trajectory

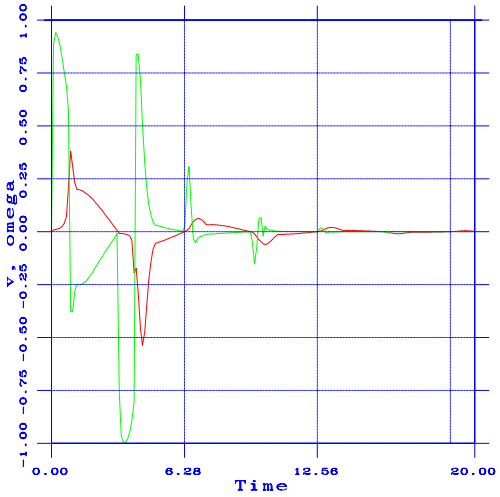

Figure 5: Mobile robot controls (v, ω)

$\frac{l_2 f}{x_m^{\{C\}}}$). The first term of equation 59 makes the arm track the targets, while the term in parenthesis pre–compensates for the motion of the mobile robot.

A useful simplification of this law is obtained by ignoring the pre–compensation term and by approximating the element $\mathcal{J}_{2,3}$ by its first term, giving finally

$$\omega_\psi(t, Y) = \lambda(Y_m - Y_{m\star}) \;/\; f \;. \tag{60}$$

This does not depend explicitly on our system's models, whose parameters may be imperfectly known, and will suffice to keep the targets in the field–of–view of the camera.

Simulations : This control law has been validated by simulations and experiments. In the simulation results presented below, we consider stabilization to zero, starting from $(x, y, \theta, \psi_1) = (0, -0.25, 0, 0)$. The gains of these controls are $k_1 = 0.25$, $k_2 = 2$, $k_3 = 100$, $\alpha = 10$, w $= 1$ and $\lambda = 1.5$ for the camera control. The mobile robot's (x, y)–trajectory is shown in fig. 4 and the controls (v, ω) are shown in fig. 5.

4.3 Simultaneous Mobile Base and Camera Pose Stabilization

The approaches in sections 4.1 and 4.2 can be seen as complementary. The first can be used to stabilize the camera to a desired position and orientation with respect to a target, but the final pose of the mobile basis is not necessarily a desired one. The second can stabilize the mobile basis to a desired pose with respect to a target and track this target with the camera while the robot moves, but it cannot independently stabilize the camera to a desired position and orientation.

If more d.o.f.s are available in the arm, they can be used to accomplish both goals simultaneously.

In this section we consider a mobile robot with a 3–d.o.f. arm as in fig. 1. Our goal is to stabilize the mobile basis to a desired pose taken as $\mathcal{X}^\star = 0$ and stabilize the camera to a desired pose determined so that the corresponding visual data are $Y_v^\star \stackrel{\text{def}}{=} (Y_l^\star,\, Y_m^\star,\, Y_r^\star)^\top$.

The system state is $X = (x, y, \theta, \psi_1, \psi_2, \psi_3)^\top$, the sensory data are $Y = (Y_l,\, Y_m\; Y_r, \psi_1, \psi_2, \psi_3)^\top$ and the controls are $\mathcal{U} = (v, \omega, \omega_{\psi_1}, \omega_{\psi_2}, \omega_{\psi_3})^\top$.

The stabilization of the mobile basis can be done as in section 4.2. The only difference is the state estimation, which is done via the matrices B_1 and B_2 that correspond to the present setup. These are given by equations 40 and 42, by setting $n = 3$.

The camera pose stabilization is cast as a regulation problem for the task function $e(X) \stackrel{\text{def}}{=} Y_v - Y_v^\star$ with dim $e = 3$, which we want to drive exponentially to zero. From 37 we get

$$\dot{e} = \dot{Y}_v = B_{2,1}(X)\; \Xi^{CT} \;. \tag{61}$$

From the system kinematics we have

$$\Xi^{CT} = \big(B_{1,1}(X)\; B_{1,2}(X)\big) B_3(X)\, \mathcal{U} \;, \tag{62}$$

where the matrices $B_{1,1},\; B_{1,2},\; B_3$ are given in 29, 30 and 45, for $n = 3$. From this and 61 and since we want to choose the arm controls so that $\dot{e} = -\lambda\, e$, we get

$$\begin{aligned}
\dot{e} &= B_{2,1}(X)\big(B_{1,1}(X)\; B_{1,2}(X)\big) B_3(X)\, \mathcal{U} \\
&= B_{2,1}(X) B_{1,1}(X) B_{3,1}(X) \begin{pmatrix} v \\ \omega \end{pmatrix} \\
&\qquad + B_{2,1}(X) B_{1,2}(X) \begin{pmatrix} \omega_{\psi_1} \\ \omega_{\psi_2} \\ \omega_{\psi_3} \end{pmatrix} \\
&= -\lambda\; e \;.
\end{aligned}$$

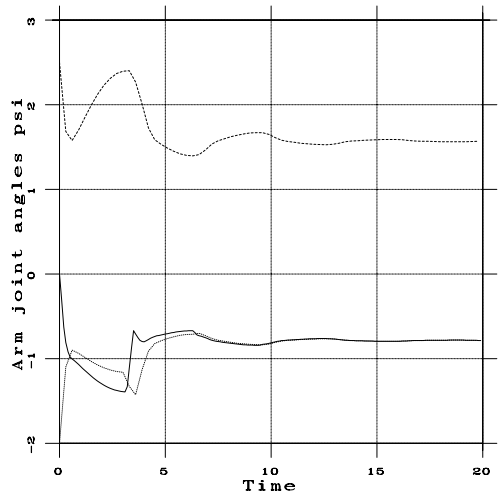

Figure 6: Arm joint angles $\psi_i,\ i = 1, 2, 3$

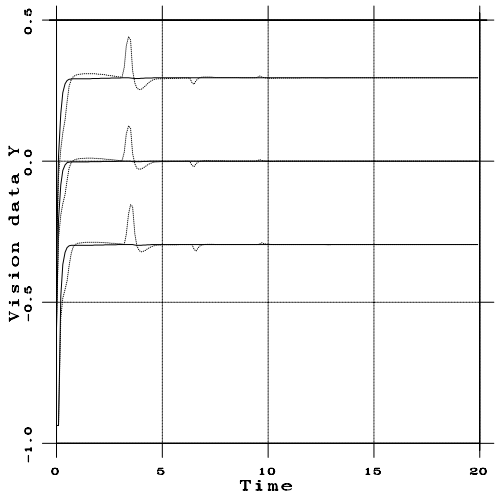

Figure 7: Visual data Y_v : with (solid) and without (dotted) precompensation

(63)

Finally, solving for the arm controls $\omega_{\psi_1},\ \omega_{\psi_2},\ \omega_{\psi_3}$ we get, away from arm and vision singularities

$$\begin{pmatrix} \omega_{\psi_1} \\ \omega_{\psi_2} \\ \omega_{\psi_3} \end{pmatrix} = -\lambda\, B_{1,2}^{-1}(X) B_{2,1}^{-1}(X)(Y_v - Y_v^\star) - B_{1,2}^{-1}(X) B_{1,1}(X) B_{3,1}(X) \begin{pmatrix} v \\ \omega \end{pmatrix}. \tag{64}$$

The first term in the R.H.S. of the above equation makes the arm track the targets, while the second term pre–compensates for the motion of the mobile basis. Notice that $\det B_{1,2} = -l_1 l_2 \sin\psi_2$, therefore configurations where it is zero are singular and should be avoided, when this control law is used.

Simulations: The feasibility of this control law has been explored by simulations, in which we consider stabilization to $(x, y, \theta, \psi_1, \psi_2, \psi_3) = (0, 0, 0, -\frac{\pi}{4}, \frac{\pi}{2}, -\frac{\pi}{4})$ starting from $(x, y, \theta, \psi_1, \psi_2, \psi_3) = (0, -0.25, 0, 0, 2.5, -2)$. The gains of these controls are $k_1 = 0.25,\ k_2 = 2,\ k_3 = 100,\ \alpha = 10,$ w $= 1$ and $\lambda = 10$ for the camera control. The mobile robot's (x, y)–trajectory is similar to the one in fig. 4 and is not shown here. The evolution of the arm joint angles ψ_i is shown in fig. 6. The convergence of the visual data Y_v to their desired values is shown in fig. 7: the solid lines correspond to the visual data obtained when the arm is controlled with the control law 64, while the dotted ones correspond to the visual data obtained when the arm is controlled without the basis motion pre–compensation in control law 64.

5 Experimental Results

The Robotic System : Our test-bed is a unicycle-type mobile robot carrying a 6 d.o.f. manipulator arm with a CCD camera and equipped with a belt of eight ultrasonic sounders (see figure 8) ([14], [15], [21]). The on-board computer architecture is built around a VME backplane. The robot controller is based on a Motorola MVME 162 with IP modules ensuring an actuator velocity servo-loop at a rate of 1ms.

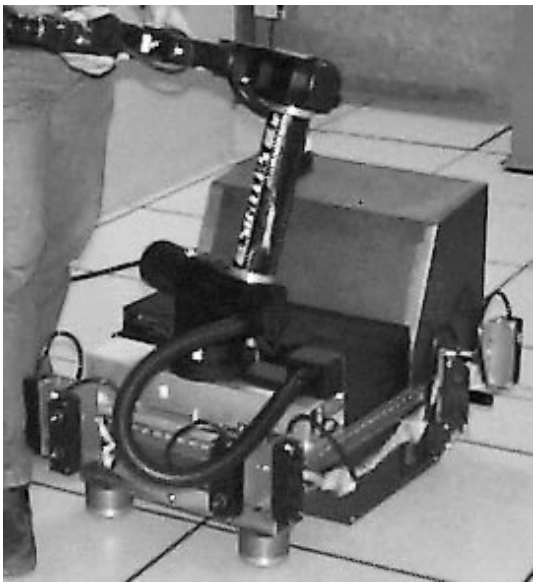

Figure 8: Mobile Manipulator

Vision Hardware and Software : To overcome difficulties in image processing, induced by strong real-

time constraints and the processing of large amounts of data, we have developed a vision machine [14] characterized by its modularity and its real-time capabilities. Its architecture is based on VLSI chips for low level processing and on DSP processors for more elaborate processing. To facilitate program development, the vision machine has been implemented on an independent VME rack outside of the mobile robot, keeping however the possibility to transfer the vision boards to the on board rack. The software used on the vision machine implements the concept of active window [14]. An active window is attached to a particular region of interest in the image and its task is to extract a desired feature in this region of interest. At each active window we associate a Kalman filter able to perform the tracking of the feature along the image sequence. Several active windows can be defined, at the same time, in the image and different types of processing can be done in each window. In our application, the visual data are extracted at video rate (see figure 9).

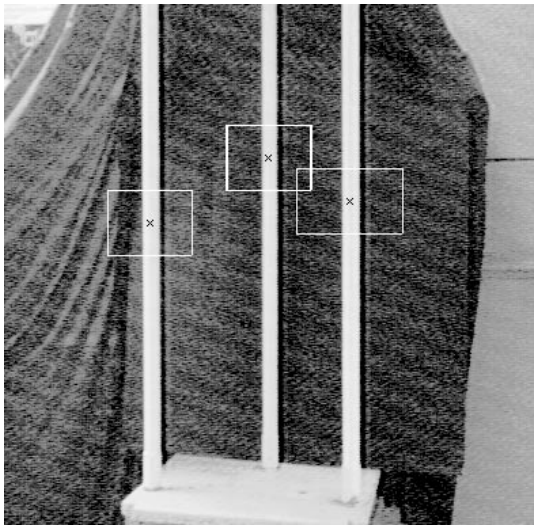

Figure 9: Active windows

Camera Pose Stabilization: The relevant experimental results were shown in section 4.1.

Mobile Base Pose Stabilization and Target Tracking Experiments: In the experimental results presented below, we use the control law 58 with the unicycle controls 54, the arm control 60 and the state approximation 57 by sensory data. The following parameters corresponding to the models developed above are used: $l_1 = 0.51\ m$, $l_2 = 0.11\ m$, $d = 2.95\ m$, $f = 1\ m$. The following gains are used for the above control laws: w $= 0.1$, $k_1 = 0.25$, $k_2 = 2$, $k_3 = 100$, $\alpha = 10$, $\lambda = 12$. The controls 55 are normalized to avoid actuator saturation and wheel sliding; this does not affect the exponential stabilization of the system, only its rate.

Initial experiments used the raw visual data to calculate the state and the controls. The resulting (x, y)–trajectory is plotted in figure 10. The dotted line represents data obtained by odometry, while the solid one represents data obtained by vision. As it is evident from this figure, implementation of this scheme leads to significant small oscillations and jerks during the motion of the system.

Subsequent experiments used Kalman filtering of each of the state variables (x, y, θ), to make the corresponding trajectories smoother and compensation of the vision–induced delays was introduced. No filtering was used on the visual data. The resulting (x, y)–trajectory and the unfiltered visual data Y_m, as well as the corresponding controls v, ω are shown in figures 11 and 12, when the system starts at $(x, y, \theta) = (-0.03, -0.23, 0.02)$. Each period of the time–varying controls corresponds to 1570 samples (data on the state of the system are recorded every 40 msec). The system motion is noticeably ameliorated [21].

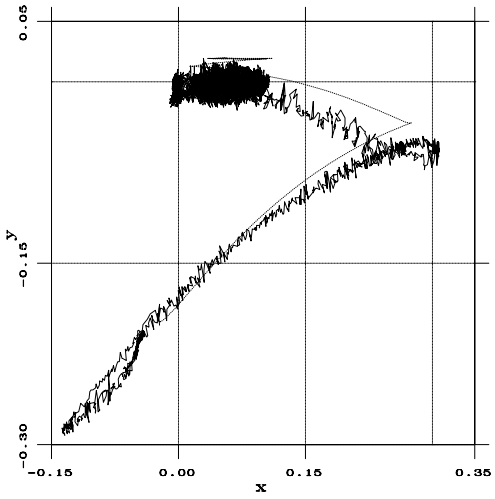

Figure 10: (x, y)–trajectory: No state filtering

6 Conclusion

We presented several approaches to the application of visual servoing techniques to systems mixing holonomic and nonholonomic mechanisms. How appropriate each of these approaches is, depends on the task to be performed and on the mechanical structure of the robot. The first approach, based on the Jacobian of the whole system, was proven sufficiently robust with respect to modeling errors and measurement noise in both simulations and experiments. In some tasks which can be uniquely specified in the camera frame (e.g. target tracking, wall following, etc.), this

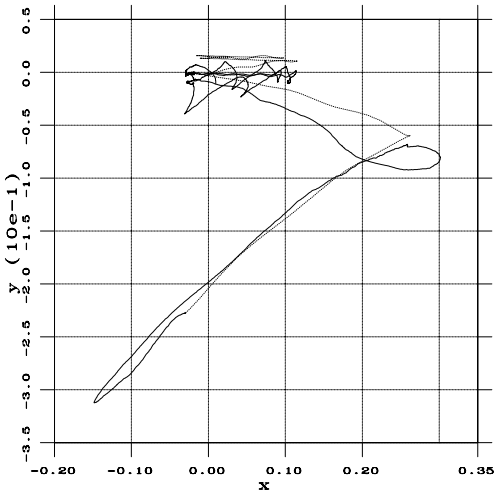

Figure 11: (x, y)–trajectory: Filtered state

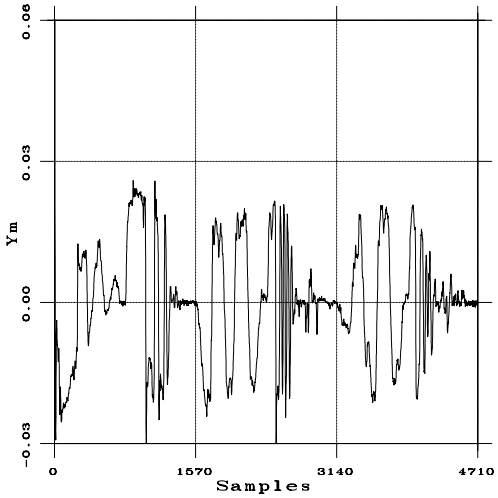

Figure 12: Visual data (Unfiltered)

first scheme will be sufficient. However, it cannot be applied anymore when the task explicitly requires the stabilization of the nonholonomic platform to a desired pose, like, for example, in a parking maneuver. Finally, the use of redundant systems allowing the simultaneous stabilization of the camera and the stabilization of the nonholonomic platform brings up some exciting research issues in a large field of applications, like those where the robot has to navigate in highly constrained environments (e.g. nuclear plants or minefields). The related results presented are however preliminary and their experimental evaluation is currently in progress using the testbed described above. In particular, several theoretical and experimental issues need to be addressed concerning the robustness of such control schemes.

References

[1] J. Barraquand, J. C. Latombe, "On nonholonomic mobile robots and optimal maneuvering", *Revue d'Intelligence Artificielle*, Vol. 3, No. 2, Hermes, pp. 77-103, 1989.

[2] R.W. Brockett, "Asymptotic Stability and Feedback Stabilization", in *Differential Geometric Control Theory*, Eds. R.W. Brockett, R.S. Millman and H.J. Sussmann, Birkhauser, Boston, 1983.

[3] C. Canudas de Wit, H. Khennouf, C. Samson, and O.J. Sordalen, "Nonlinear control design for mobile robots", pp. 121-154, in *Recent trends in mobile robots* , Ed. Y.F. Zheng, Series in Robotics and Automated Systems, Vol. 11, World Scientific, 1994.

[4] F. Chaumette, *La relation vision-commande: théorie et applications à des tâches robotiques*, Ph.D. Thesis, University of Rennes I, France, July 1990.

[5] J.–M. Coron, "On the Stabilization in Finite Time of Locally Controllable Systems by Means of Continuous Time–Varying Feedback Law", *SIAM J. Control and Optimization* **33**, 804-833, 1995.

[6] B. Espiau, F. Chaumette and P. Rives, "A New Approach to Visual Servoing in Robotics", *IEEE Trans. on Robotics and Automation* **8**, 313-326, 1992.

[7] G.D. Hager and S. Hutchinson, Eds., "Vision–based Control of Robotic Manipulators", Special section of *IEEE Trans. Robotics and Automation* **12**, 649-774, 1996.

[8] K. Hashimoto, Ed., *Visual Servoing*, World Scientific, 1993.

[9] B.K.P. Horn, *Robot Vision*, Mc Graw–Hill, 1986.

[10] P. Morin and C. Samson, "Application of Backstepping Techniques to the Time–Varying Exponential Stabilization of Chained Form Systems", INRIA Research Report No. 2792, Sophia–Antipolis, 1996 (Also submitted to *European Journal of Control*).

[11] R.M. Murray, Z. Li and S.S. Sastry, *A Mathematical Introduction to Robotic Manipulation*, CRC Press, Boca Raton, 1994.

[12] R. Pissard–Gibollet and P. Rives, "Applying Visual Servoing Techniques to Control a Mobile Hand–Eye System", *IEEE Intl. Conf. on Robotics and Automation*, 1995.

[13] J.–B. Pomet and C. Samson, "Time–Varying Exponential Stabilization of Nonholonomic Systems in Power Form", INRIA Research Report No. 2126, Sophia–Antipolis, 1993.

[14] P. Rives, J.J. Borrelly, J. Gallice and P. Martinet, "A Versatile Parallel Architecture for Vision Based Applications", Workshop on Computer Architecture for Machine Perception, New Orleans, 1993.

[15] P. Rives, R. Pissard-Gibollet and K. Kapellos, "Development of a Reactive Mobile Robot Using Real–Time Vision", Third International Symposium on Experimental Robotics, Kyoto, Japan, October 28-30, 1993.

[16] P. Rives, R. Pissard-Gibollet and L. Pelletier, "Sensor-based Tasks: From the Specification to the Control Aspects", The 6th Intl. Symposium on Robotics and Manufacturing, Montpellier, France, May 28-30, 1996.

[17] C. Samson, M. Le Borgne and B. Espiau, *Robot Control: The Task Function Approach*, Oxford University Press, 1991.

[18] C. Samson, "Velocity and Torque Feedback Control of a Nonholonomic Cart", in *Advanced Robot Control*, Ed. C. Canudas de Wit, Lecture Notes in Control and Information Sciences, No. 162, Springer-Verlag, 1990.

[19] D.P. Tsakiris, C. Samson and P. Rives, "Vision–based Time–varying Mobile Robot Control", Final European Robotics Network (ERNET) Workshop, Darmstadt, Germany, September 9–10, 1996. Published in *Advances in Robotics: The ERNET Perspective*, Eds. C. Bonivento, C. Melchiorri and H. Tolle, pp. 163-172, World Scientific Publishing Co., 1996.

[20] D.P. Tsakiris, C. Samson and P. Rives, "Vision–based Time–varying Stabilization of a Mobile Manipulator", Proceedings of the Fourth International Conference on Control, Automation, Robotics and Vision *(ICARCV'96)*, pp. 2212-2216, Westin Stamford, Singapore, December 4–6, 1996.

[21] D.P. Tsakiris, K. Kapellos, C. Samson, P. Rives and J.–J. Borrelly, "Experiments in Real–time Vision–based Point Stabilization of a Nonholonomic Mobile Manipulator", Preprints of the Fifth International Symposium on Experimental Robotics (ISER'97), pp. 463-474, Barcelona, Spain, June 15–18, 1997.