Head-Centered Orientation Strategies in Animate Vision

Enrico Grosso* and Dana H. Ballard

Department of Computer Science
University of Rochester
Rochester NY 14627-0226

Abstract

This paper is about orienting, *that is, establishing and maintaining a spatial relation between a motorized pair of cameras (the eye-head system) and a static or a moving object tracked over time. Motivated by physiological evidence, the paper proposes a simple set of vision-based strategies aimed to perform head, eyes and body movements in a complex environment. Fixation is shown to be an essential feature in visual servoing, and it is used to decouple control on head rotational degrees of freedom, making possible a metric-less approach to the orientation problem. A running implementation of these strategies, using a binocular camera system mounted on a PUMA 700, demonstrates the effectiveness of the approach.*

Key words: vision; robotics; visual orientation; head-eye coordination.

1 Vision and orientation

Complex visual tasks usually require the establishment of a consistent and stable relation with the objects involved in the task. A simple example is sketched in figure 1: to accomplish the task of reading the book the observer should change its current relation with the book in a useful way; typically, the observer should be nearer to the book and properly oriented with respect to the surface of the pages. We think this example is indicative of a general case, that of a complex task that has the following embedded component:

> establish and maintain the most convenient spatial orientation in relation to the task and to the objects involved in the task

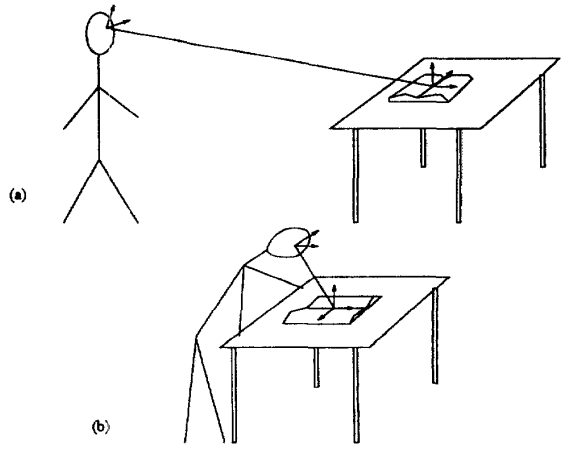

Figure 1: Task-dependent spatial orientation is required to "read the book".

The example of orienting in order to read a book reveals some important properties. First of all, the orientation sub-task is by definition an *ongoing process*; in a time-varying world the spatial orientation has to be maintained continuously. Secondly, the spatial orientation is *task-dependent*; certainly the approach used to read a book differs from the approach used to type on a workstation or to grasp a teapot.

One way to solve this problem would be to use metric vision to measure the position of the book with respect to the observer and then program, possibly in one shot, the movement required to gain a useful orientation. However, this approach is difficult for at least two reasons. First, it requires calibration of the cameras, and that is usually difficult even when good reference points are available; second, even if the system is able to track multiple points without errors, depth information computed using traditional methods, like

*E. Grosso was a visiting scientist from the Department of Communication, Computer and Systems Science, University of Genoa, Italy. This work has been supported by a grant of the Italian National Council of Research and grants from NIH (No. 1R24RRO6853) and the Human Scientific Frontiers Program.

0-8186-3870-2/93 $3.00

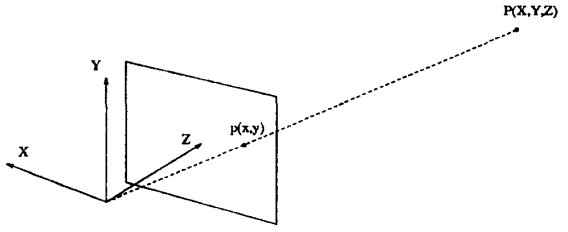

Figure 2: Pinhole model of the camera system.

stereo or motion analysis, is not always accurate, and sometimes it is not available.

An alternative approach to the problem is based on continuous visual servoing [3]. In this case the positioning task is defined in terms of image features and a real-time control loop is used to modify the observer position toward the desired one. More analytically, referring to figure 2, the inverse perspective transform relates the position of the image point $\vec{p} = (x, y)$ as a function of the space point $\vec{P} = (X, Y, Z)$:

$$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} X \\ Y \end{bmatrix} \cdot \frac{F}{Z} \tag{1}$$

where F is the focal length of the adopted pinhole model.

Moving the imaging system with respect to the space point generates a displacement of the point $\vec{p}$ into the image plane; the velocity, or optical flow, of the point $\vec{p}$ can be expressed in terms of the observer velocity differentiating equation (1) with respect to time. This leads to the well known relation [6]:

$$\begin{bmatrix} \dot{x} \\ \dot{y} \end{bmatrix} = \begin{bmatrix} \frac{-F}{Z} & 0 & \frac{x}{Z} & \frac{xy}{F} & \frac{-(F^2+x^2)}{F} & y \\ 0 & \frac{-F}{Z} & \frac{y}{Z} & \frac{(F^2+x^2)}{F} & \frac{-xy}{F} & -x \end{bmatrix} \begin{bmatrix} \dot{X} \\ \dot{Y} \\ \dot{Z} \\ \dot{\omega}_x \\ \dot{\omega}_y \\ \dot{\omega}_z \end{bmatrix} \tag{2}$$

Using a sufficient number n of image points and inverting equation (2) (we do not discuss here the conditions that make this possible but it is obvious that degenerate cases can occur) allows the control vector $[\dot{X}\ \dot{Y}\ \dot{Z}\ \dot{\omega}_x\ \dot{\omega}_y\ \dot{\omega}_z]^T$ to be expressed as a function of image velocities $[\dot{x}_1\ \dot{y}_1\ \dot{x}_2\ \dot{y}_2 \cdots \dot{x}_n\ \dot{y}_n]^T$. In this way one can design a control algorithm to make image points converge toward the desired features and to move the observer, in the same time, toward the desired spatial position. One major drawback of this approach is the computational complexity of the control algorithm; the matrix to be inverted depends on the positions of the image points and must be re-estimated at each step. Moreover, a calibration phase is required to estimate focal length and pixel dimensions, and Z distance of points in space must be known or estimated in some way.

To avoid the above difficulties, we propose a different approach to continuous visual servoing; in particular, we describe how we can fruitfully exploit gaze control and a relative metric to plan changes of orientation.

1.1 Fixation

Fixation (or *gaze control*) is a common behavior in mammals and lower vertebrates [8]; it confers advantages in vision that are especially useful for humans and other species provided by foveal vision:

- the permanence of a selected target in the foveal region allows the visual process to best exploit the high resolution available in this area [7];
- gaze holding capabilities can be used to track objects without regard to object recognition; on the contrary, when object or observer movements are involved in the fixation process, they can be useful to better isolate (by motion blur) the tracked object [1];
- an active following of the target allows the use of relative coordinate systems; this simplifies recognition (if required) and the spatial description of the target itself.

Fixation is the only simple way to couple proprioceptive sensors (observer reference system) and external space (object reference system). Referring to figure 3, imagine the movement of a single camera, following a moving target in space. In case (a) the camera is able to maintain fixation on the target and the rotation of the camera (let us say the rotation perceived by the observer) is a direct measure of the change in orientation with respect to the target itself. In case (b) the camera is not able to maintain a perfect fixation and the final result is that the rotation perceived by the observer does not correspond to the change in orientation. To calculate the remaining angle we need to know intrinsic camera parameters. Thus, while maintaining fixation it is easy to maintain coherent spatial relationships; losing fixation makes the problem much more difficult, requiring traditional metric techniques.

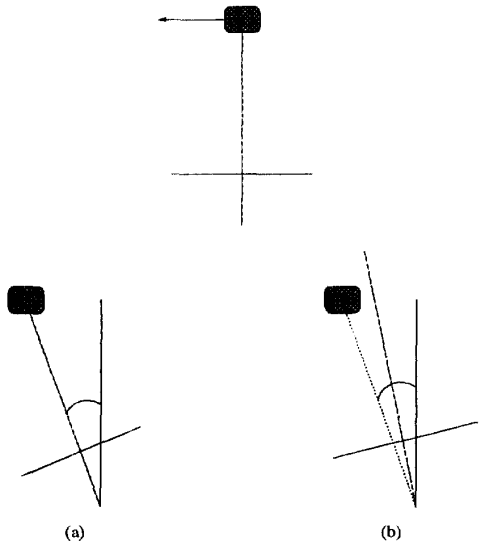

Figure 3: A moving object is tracked by a moving camera: (a) perfect and (b) non-perfect gaze control.

The effect of fixation may be understood in terms of equation (2). The fixation constraint is:

$$\begin{bmatrix} 0 \\ 0 \end{bmatrix} = \begin{bmatrix} \frac{-F}{Z_{fp}} & 0 & 0 & 0 & -F & 0 \\ 0 & \frac{-F}{Z_{fp}} & 0 & F & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} \dot{X} \\ \dot{Y} \\ \dot{Z} \\ \dot{\omega}_x \\ \dot{\omega}_y \\ \dot{\omega}_z \end{bmatrix} \quad (3)$$

Solving for $\dot{\omega}_x$ and $\dot{\omega}_y$ and substituting in (2) for a generic point:

$$\begin{bmatrix} \dot{x} \\ \dot{y} \end{bmatrix} = \begin{bmatrix} \frac{x^2}{F \cdot Z_{fp}} + F\frac{Z - Z_{fp}}{Z \cdot Z_{fp}} & \frac{xy}{F \cdot Z_{fp}} & \frac{x}{Z} & y \\ \frac{xy}{F \cdot Z_{fp}} & \frac{y^2}{F \cdot Z_{fp}} + F\frac{Z - Z_{fp}}{Z \cdot Z_{fp}} & \frac{y}{Z} & -x \end{bmatrix} \begin{bmatrix} \dot{X} \\ \dot{Y} \\ \dot{Z} \\ \dot{\omega}_z \end{bmatrix} \quad (4)$$

Equation (4) still involves camera parameters and distance information but, remarkably, it provides a complete decoupling of $\dot{\omega}_x$ and $\dot{\omega}_y$. In other words, since the rotations around the x and y axes are constrained to maintain fixation, the displacement of the image points is completely determined by the remaining four degrees of freedom. It is important to emphasize that equation (4) is based on differential geometry; to achieve large changes in orientation we must use it within a continuous servo loop.

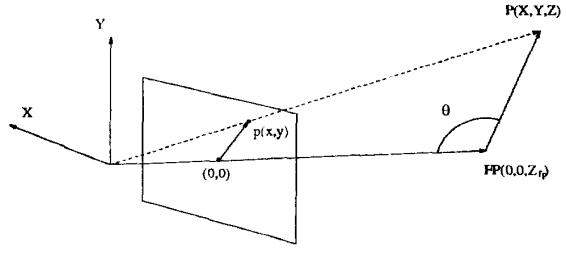

Figure 4: The orientation with respect to a vector in space is defined by the θ angle.

1.2 Controlling the orientation of the optical axis

To simply change the *orientation of the optical axis* with respect to a vector in space, it is easy to verify that both $\dot{Z}$ and $\dot{\omega}_z$ are ineffective. In this case, then, we only have to deal with the first part of equation (4):

$$\begin{bmatrix} \dot{x} \\ \dot{y} \end{bmatrix} = \begin{bmatrix} \frac{x^2}{F \cdot Z_{fp}} + F\frac{Z - Z_{fp}}{Z \cdot Z_{fp}} & \frac{xy}{F \cdot Z_{fp}} \\ \frac{xy}{F \cdot Z_{fp}} & \frac{y^2}{F \cdot Z_{fp}} + F\frac{Z - Z_{fp}}{Z \cdot Z_{fp}} \end{bmatrix} \begin{bmatrix} \dot{X} \\ \dot{Y} \end{bmatrix} \quad (5)$$

Referring to figure 4, the best way to modify the angle θ between the vector $(\vec{P} - \vec{FP}) = (X, Y, Z - Z_{fp})$ and the optical axis is to program a movement in the direction of $\vec{p} = (x, y)$. Denoting by Δh a small displacement we have:

$$\Delta X = \Delta h \cdot x \qquad \Delta Y = \Delta h \cdot y$$

and, substituting in equation (5):

$$\frac{\Delta x}{\Delta y} = \frac{x}{y}$$

In other words, the projection of point P will move along the direction of $\vec{p}$ and the angle θ will change in accordance with the camera movement. If the task requires a more quantitative approach (that is, we need an estimate of the angle θ) then we have to use relative depth; in fact we can derive:

$$\cos\theta = \frac{Z_{fp} - Z}{\sqrt{X^2 + Y^2 + (Z - Z_{fp})^2}}$$

or, using the perspective transform:

$$\cos\theta = \frac{1 - \frac{Z}{Z_{fp}}}{\sqrt{\frac{Z^2}{Z_{fp}^2} \cdot \frac{x^2}{F^2} + \frac{Z^2}{Z_{fp}^2} \cdot \frac{y^2}{F^2} + \left(1 - \frac{Z}{Z_{fp}}\right)^2}}$$

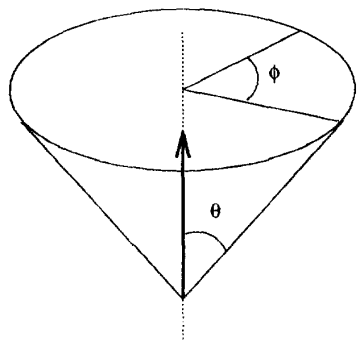

Figure 5: Rotation around a vector in space is defined by the ϕ angle.

Similarly, it is easy to verify that a movement in the direction perpendicular to $\vec{p}$ will produce an image flow instantaneously perpendicular to $\vec{p}$:

$$\Delta X = \Delta h \cdot y \qquad \Delta Y = -\Delta h \cdot x$$

$$\frac{\Delta x}{\Delta y} = -\frac{y}{x}$$

During the movement, the optical axis will rotate around the space vector $(\vec{P} - \vec{FP})$, the angle ϕ will change, and the angle θ will remain unaffected, as shown in figure 5. A measure of the angle ϕ can be derived in this case only in relation to a second vector in space; basic geometry shows that the *oriented angle* between the projection of the two vectors univocally determines the spatial position of the optical axis.

The consequences of the equations above are extremely important. They provide a general method to control the position of the optical axis in space, which is independent of the values of $\dot{Z}$ and $\dot{\omega}_z$. The method is based on simple translations, in a plane parallel to the image plane, and it requires only three points in space (two vectors), one of which is the fixation point. Again, we emphasize that this analysis is differential, and that fixation and translations must be performed in continuous closed loop.

1.3 Controlling the radial degrees of freedom

There is an almost direct relationship between this approach and the work of Koenderink [5, 4] on optical flow. Koenderink has shown that the motion parallax field is related to the slant of the surface with respect to the observer, and has proposed a differential decomposition of this field in three parts: divergence (change of the area projected into the image), curl (rotation of the image around the optical axis) and deformation (expansion or compression along a specific image direction). Denoting by $\vec{n}$ the normal to the surface and by $\vec{n}_i$ the projection of $\vec{n}$ into the image plane, the divergence, curl and deformation are defined as:

$$\begin{aligned} div &= K\ (\vec{n}_i \cdot (\dot{X}, \dot{Y}) + 2\dot{Z}) \\ curl &= K\ (\vec{n}_i \times (\dot{X}, \dot{Y}) - 2\dot{\omega}_z) \\ def &= K\ (|\vec{n}_i| \cdot |\dot{X}, \dot{Y}|) \end{aligned} \tag{6}$$

where K is a constant factor. The meaning of equations (6) can be related to our control problem in the following way. Consider a generic space vector as the normal to a small piece of surface. It turns out that a movement aimed to change the angle θ will produce divergent flow and deformation but not curl. In the same way a movement aimed to change the angle ϕ will produce curl and deformation but not divergence.

In turn, the two parameters $\dot{Z}$ and $\dot{\omega}_z$ can be related to the independent control of curl and divergence, without producing undesired deformations. Also, note that $\dot{Z}$ does not affect at all the orientation between the observer reference frame and the object reference frame. Control of $\dot{Z}$ can run a completely independent servo loop, and it is essentially related to the control of the resolution of image information during the execution of the task (reading a book is a classical example); a positive $\dot{Z}$ will expand the image uniformly while a negative $\dot{Z}$ will produce the opposite effect.

In contrast, imposing constraints on ω_z limits, in general, the orientation capabilities of the observer. Even though the quality of image information does not change by rotating around the optical axis, orientation errors about this axis can have considerable effects on the processing capabilities of visual information. In other words, we can put a constraint on ω_z but we also have to implement a relaxation mechanism for this constraint, such that the system will maintain a preferred orientation but will be able, when necessary, to change to a more convenient one. It is a common experience that humans, during generic tasks execution, maintain a coherent position with respect to the gravity field. This is probably due to inertial cues; the gravity field completely embeds our world, and it is not surprising that it usually constrains human movements. The hypothesis we propose here is that, depending on the rotations ω_x and ω_y, the observer system rotates in turn around the Z axis to maintain the X axis parallel to the ground (see figure 6 for details). This corresponds, in practice, to constraining the rotational movements of the observer around two

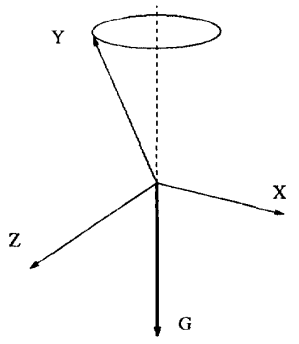

Figure 6: Relation between the observer reference system and the gravity vector.

directions: the X axis (always perpendicular to $\vec{G}$) and the absolute vertical direction (indicated by $\vec{G}$).

Writing explicitly the relations among ω_x, ω_y, ω_z and $\vec{G} = (0, g_y, g_z)$ and differentiating ω_z we have:

$$\dot{\omega}_z = \frac{g_z}{g_y}\dot{\omega}_y = -\frac{g_z}{g_y} \cdot \frac{\dot{X}}{Z_{fp}}$$

As expected, $\dot{\omega}_z$ is simply a function of $\dot{\omega}_y$ and $\vec{G}$; due to the fixation constraint, it depends equivalently on $\dot{X}$ and $\vec{G}$.

In practice, $\dot{\omega}_z$ can run an independent control loop, processing inertial signals (to detect the gravity vector) and using image information (the projection of a space vector is sufficient) to relax the gravity constraint.

2 A practical implementation

A complete implementation of the issues described in this paper would require remarkable parallel processing capabilities, fast visual processing and fast controllers for the mechanical parts. We present a preliminary implementation, in which eight control loops work in parallel; the system is able to track actively a single target moving in space, compensating for camera rotation and maintaining, by a servo loop on $\dot{Z}$, a constant distance from the target. Translational movements along the X, Y and Z axes of the head can be directly controlled by the user; also in this case the cameras and the head will rotate automatically to track and to compensate in real time. In the following, the system is presented as a whole and a brief discussion is provided in terms of global functionalities; then, the implementation of the single sub-parts is described and some implementative issues are discussed.

2.1 System overview

Head and eyes differ significantly in dynamics; the rotational inertia of the head has been estimated to be about 10^4 times the inertia of the eyes while viscosity differs by a scaling factor of about 10^2 [9]. These differences explain clearly some common behaviors, like the fast movements of the eyes during a saccade, followed by a slower compensation of the head.

Taking into account physiological data is not always easy in robotics and sometimes it requires considerable simplifications. Figure 7 depicts a general framework derived from physiological evidence but aimed toward an artificial implementation. The innermost level comprises the eye sub-system; it is responsible for tracking the target in real time, modifying opportunely both vergence and tilt angles. In our implementation, the eye sub-system is realized by means of a couple of CCD cameras driven by three separate motors [2] and is intended to be the fastest subsystem among those implemented. At the intermediate level there is the head sub-system, providing compensatory movements for the eyes in two directions (tilt and pan); the implementation is realized using the PUMA wrist. At the outermost level there is the body sub-system; using the PUMA arm it provides three translational degrees of freedom. In the current implementation, the body sub-system does not provide rotational compensation for the head, but this issue could be very important in more complex implementations.

2.2 The separate control loops

The complete control system consists of eight separate processes that are described as follows:

(1) Left camera (eye) pan.
The goal is to follow the selected target; control is based on visual information.

(2) Right camera (eye) pan.
As for the left camera; control is based on visual information.

(3) Cameras common tilt.
The goal is to follow the selected target; control is based on visual information.

(4) Head tilt (rotation around the X axis).
The goal is to compensate for cameras tilt; con-

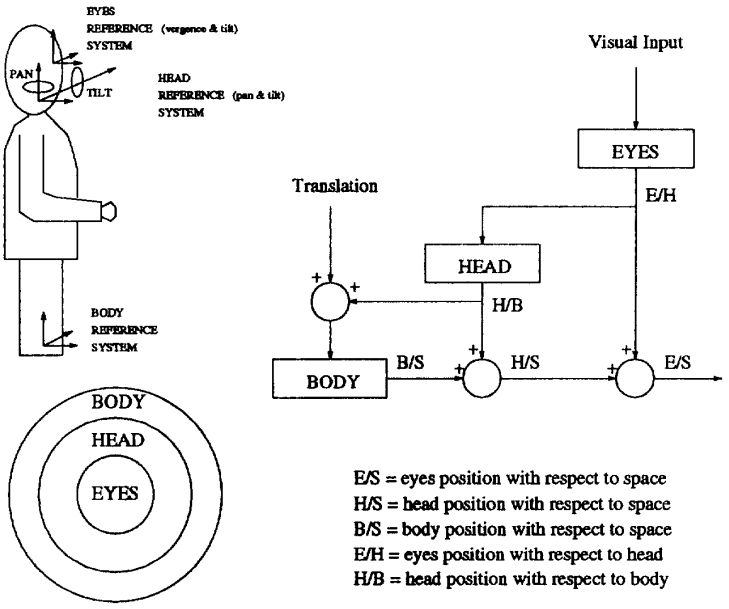

Figure 7: A general framework for the eyes-head-body system.

trol is based on information coming from motor encoders.

(5) Head pan (rotation around the Y axis).
The goal is to compensate for cameras pan; control is based on information coming from motor encoders.

(6) Head translation along the X axis.
It is activated only by voluntary movements, aimed to change the orientation of the optical axis, at this stage user commands.

(7) Head translation along the Y axis.
It is activated only by voluntary movements, aimed to change the orientation of the optical axis, at this stage user commands.

(8) Head translation along the Z axis.
The goal is to maintain the target at a fixed distance; control is based on the area of the target projected into the image. User commands can be superimposed on the basic control loop, modifying the desired distance.

Finally, it is important to emphasize that the actual implementation so far does not completely perform automatic positioning. In particular, the visual processing provides tracking of a single target, while three different targets need to be simultaneously tracked for a real positioning task. Moreover, ω_z is actually not used; again, this is due to the limitations in visual processing (at least two points are required), and to the fact that inertial sensors are still unavailable for the system. Regardless of these aspects, however, the system is sufficiently responsive (the maximum cycle time for a process is about 300 ms.) and it already contains all the basic features that could be extended in future implementations.

Figures 8 and 9 show some pictures extracted from two different experimental sequences. In the first sequence the system runs freely (the user does not command translations) and the target is moved up and down, at steps of about 150mm., to check the response of the servo loops (1) to (5) and, in particular, (8). Figure 10 shows the effect of the robot motion on the rightmost image; the change of the projected area activates the servo loop controlling the Z axis and, in turn, all the rotational degrees of freedom. In the plot four different steps are clearly visible (two backward and two forward), as is the effect of robot motion in restoring the original distance from the target and the original area in the image.

In the second sequence the user commands a movement along the y axis, first "down" and then "up"; the resulting robot trajectory is a curve around the tar-

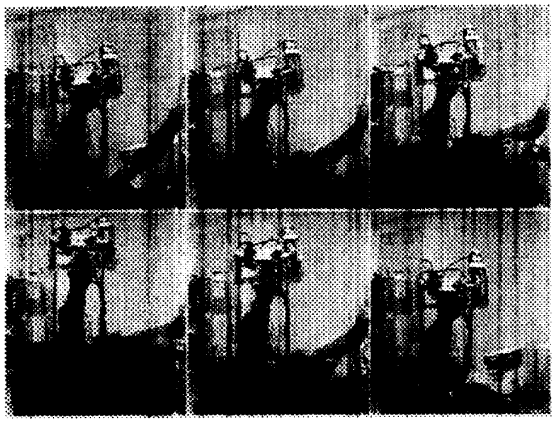

Figure 8: Sequence 1: when the target is moved up and down, the system moves to compensate for the area change.

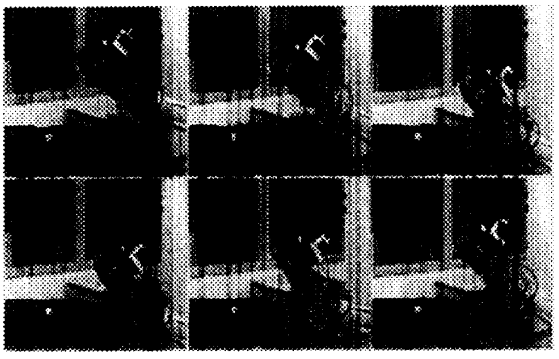

Figure 9: Sequence 2: due to the fixation constraint and to the depth servo loop, a translational command generates a curved trajectory.

get, maintaining approximately a fixed distance from the target. Figures 12 and 11 show the values of the main image features during the motion: in particular, the x and y coordinates of the target are plotted for the right camera, together with the projected area. As expected, the commanded motion particularly affects the y coordinate and the area of the target projection, while the servo loop controlling the x coordinate only compensates for small fluctuations.

3 Conclusions

We propose a new approach to the orientation control of a robotic system. The approach is based on

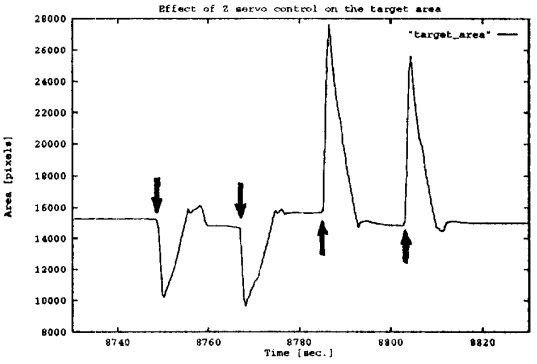

Figure 10: Sequence 1: the behavior of the system in response to step changes in depth (arrows) is shown using the area of the target projection.

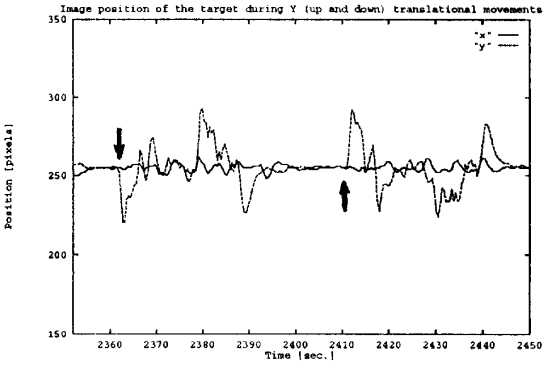

Figure 11: Sequence 2: the behavior of the system in response to translational commands down and up (arrows) is shown using the position of the gravity center of the target.

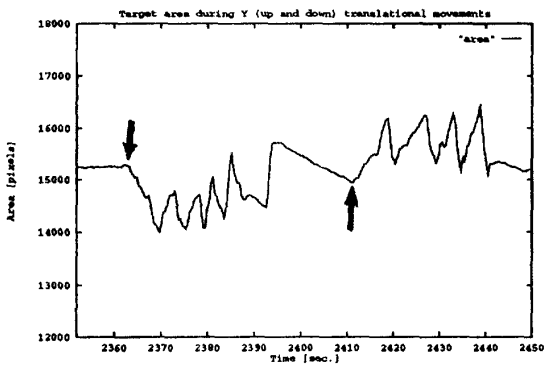

Figure 12: Sequence 2: the behavior of the system in response to translational commands down and up (arrows) is shown using the area of the target projection. Jerky movements reflect limitations in the PUMA control system.

multiple independent processes, controlling different degrees of freedom of an eye-head system, and running in parallel and in real time. Visual information is essentially used to control the orientation strategy; proprioceptive measures are also exploited for head compensation. A preliminary implementation is shown, in which eight degrees of freedom are controlled in parallel. Future work will be devoted to the integration of inertial sensors and to the implementation of more powerful visual processing capabilities.

References

[1] D.H. Ballard, R.C. Nelson, and B. Yamauchi. Animate vision. *Optics News*, 15(5):17–25, 1989.

[2] C.M. Brown. The Rochester robot. Technical Report TR 257, University of Rochester, 1988.

[3] B. Espiau, F. Chaumette, and P. Rives. A new approach to visual servoing in robotics. *IEEE Transactions on Robotics and Automation*, 8(3):313–326, June 1992.

[4] J.J. Koenderink. Optic flow. *Vision Research*, 26(1):161–180, 1986.

[5] J.J. Koenderink and A.J. van Doorn. Invariant properties of the motion parallax field due to the movement of rigid bodies relative to an observer. *Optica Acta*, 22(9):773–791, 1975.

[6] J.H. Rieger and D.T. Lawton. Processing differential image motion. Technical Report COINS TR 84-28, University of Massachusetts, Amherst, 1984.

[7] G. Sandini and P. Dario. Active vision based on space-variant sensing. In *Proc. 5th Int. Symposium on Robotics Research*, Tokyo, Japan, 1989. MIT Press.

[8] R.H. Schor, R.E. Kearney, and N. Dieringer. Reflex stabilization of the head. In B.W. Peterson and F.J. Richmond, editors, *Control of Head Movement*, pages 141–166. Oxford University Press, 1988.

[9] L. Stark, W.H. Zangemeister, and B. Hannaford. Head movement models, optimal control theory, and clinical application. In B.W. Peterson and F.J. Richmond, editors, *Control of Head Movement*, pages 245–260. Oxford University Press, 1988.