Gaze Control Using Human Eye Movements

Fabien Spindler, François Chaumette

IRISA / INRIA Rennes
Campus universitaire de Beaulieu, 35042 Rennes Cedex, France
Email {fspindle, chaumett}@irisa.fr

Abstract

In this paper, we propose a visual servoing of the orientation of a camera mounted on a robot, using human eye movements. The aimed domain is the remote survey of sensitive sites or the intervention in contaminated nuclear sites. Visual data stemmed from an eye-tracker, providing the direction of the look of an operator, is generally noisy and sometimes inconsistent. Filtering, smoothing and stabilizing this eye-tracker data are thus necessary in order to move the camera in the direction specified by the operator's eye. The visual servoing control scheme, as well as experimental results, are also presented.

1 Introduction

In vision-based control [1] [2] [4] [6] [8] [9], the main point is to be able to act on the environment or to supervise the operation process. It implies a special visual or algorithmic attention on one or several image regions of interest. Visual data used in visual servoing is generally extracted by quite simple image processing algorithms, due to the high rate needed for the stability and robustness of the control schemes. For example, very simple white objets on black background are generally considered. Let us note however that significant improvements are in progress [5].

Dealing with our application which aimed domain is the remote survey of sensitive sites or the intervening in contaminated nuclear sites, we propose a visual servoing of the orientation of a camera mounted on a robot by using human eye movements. To have the same kind of behavior as the human eye, only camera pan and tilt are controlled.

Using an eye-tracker makes it possible to determine the gazed direction of a human operator. After a simple calibration step, it is possible to translate this gazed direction into pixel coordinates corresponding to the image area focused on by the operator. However, these coordinates are quite noisy - due for example to ambient lighting or to aberrant data induced by flickers of the eyelids. So, these coordinates must be filtered, smoothed and stabilized in order to obtain precisely the center of gravity of the focused region. Integrating these coordinates into a control law allows us to achieve human controlled focusing tasks. In addition, we have placed a controller system of the camera zoom-lens at the operator's disposal.

The next section of this paper presents the experimental system used to realize the gaze control using human eye movements. Section 3 recalls the visual data extraction using the eye-tracker and its filtering. Section 4 describes the application of the task function approach to visual servoing and the expression of the resulting control law. We finally present real time experimental results dealing with focusing tasks (with respect to gazed points by an operator), as well as target tracking.

2 Experimental system

The structure of the whole system is shown in Figure 1. Images captured by the camera mounted on the end effector of a robot are sent to the screen observed by the operator. The TV screen reproducing the camera picture has a 55 cm diagonal and is 1.50 m far from the eye-tracker. The eye-tracker calculates the coordinates focused by the observer and sends them by a RS-232C serial port to a SPARC Station 20. The host machine computes the gaze control law using the eye-tracker outputs and sends the pan and tilt velocities to the robot controller via a Sbus-VMEbus adapter. It also calculates control inputs for the zoom mechanism, sent via a RS-232C serial port to the camera module. Classical C language has been used to implement the corresponding software.

Figure 2 shows the camera-manipulator system

used for experiments. The robot is a 4 degrees of freedom AFMA ROBOTS cylindrical manipulator. Only pan and tilt are used for the gaze control. At the end effector of the robot, a CCD SONY camera is mounted. A lens SERVOLENS X10 (from 10 *mm* till 100 *mm* focal length) is attached to the camera. Zoom, focus and iris of this lens can be controlled via a RS-232C serial port.

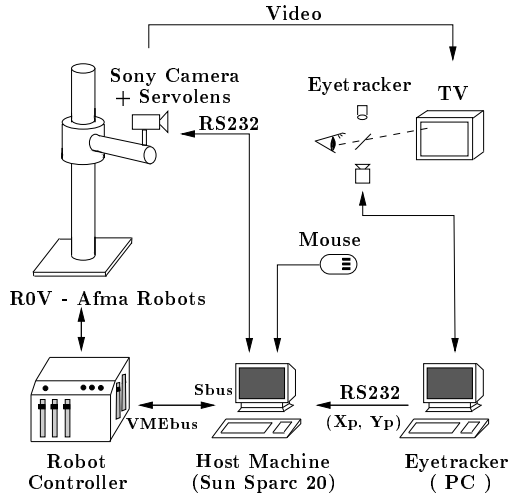


Figure 1: Experimental setup.

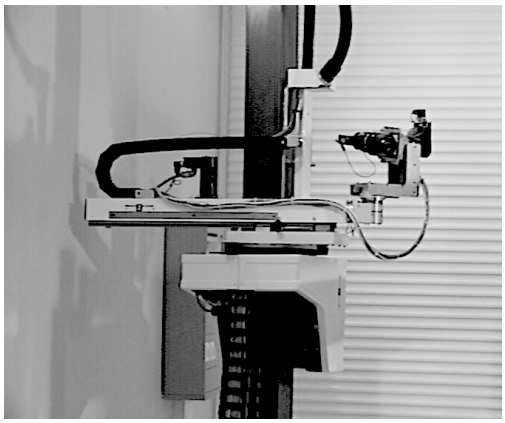


Figure 2: Experimental cell (camera mounted on a 4 d.o.f. robot).

To avoid saturating the CCD matrix, the auto-iris mode of the lens is activated. The focus is ajusted at the beginning of the experiments. On the other hand, the operator may wish to achieve details of a global scene. For such operations, the operator can stimulate the zoom-lens positioning by using a mouse in order to increase or decrease the focal length of the lens. After detecting the button click event, SERVOLENS focal length is controlled by a serial port (9600 bauds) by constant increase (ZOOM +) or decrease (ZOOM -) of 4 *mm*.

In visual servoing, more and more applications include the automatic zoom-lens control [3] [7], where the surface of an object must be kept constant in the image. In our case, since the image processing is only founded on the position of the image area gazed by the operator's eye, these techniques cannot be applied.

3 Visual data extraction

3.1 The eye-tracker

An eye-tracker is a device able to measure the direction of the operator's look. This direction can be computed with respect to the operator's head or a fixed reference frame. In that last case, head motion must be taken into account.

The first solution to take into account these head movements is to add a camera which is fixed on the operator's head. Thus, it is constantly possible to link the gaze direction to a point in the image of the scene. This solution leaves the operator free to move.

Another solution, corresponding to the one used in our experiments, is the suppression of these movements by making the head rigid with respect to the scene (by using a fixed mechanical system). The disadvantage of this system is that the user is not free. However, this drawback is not prejudicial in the case of measures made in a restricted field of view, which is our case since the observer is located in front of a TV screen.

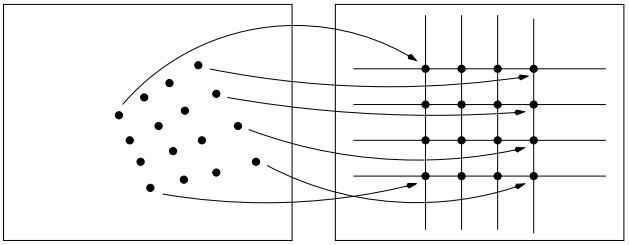
Our eye-tracker (EYEPUTER system developed by LETI in Grenoble and marketed by Alpha Bio Technologies) is built on a standard set of acquisition and image processing techniques. Figure 3 shows the EYEPUTER system, composed of an optical system, a CCD camera and two infra-red sources. The whole is held by a support fixed on a table to prevent the head from moving. A camera acquires at video rate images of the pilot's eye which is alternately lighted by one of the infra-red sources. Computations are performed on a classical PC 486.

When using the eye-tracker, two different stages must be distinguished:

- **the visual data extraction;** The EYEPUTER software computes in real-time the gaze direction from the measures of cornea reflections positions and pupil positions in the eye's image. To be less sensitive to interferences due to lighting variations, a noise elimination is also made. Thus, the eye-tracker restores the focused coordinates in the image at 15 Hz and transmits them to the host machine via a RS-232C serial port;

Figure 3: Experimental eye-tracker.

- **an off-line calibration;** In order to obtain the gazed coordinates in pixel coordinates (X_p, Y_p) in the image seen by the operator, an off-line calibration is necessary. It consists in making a comparison between the eye-tracker measurements (see Figure 4.a) and the focused intersection of a 16 points grid, projected on the screen in front of the human pilot (see Figure 4.b). The memorizing of these deviations makes possible to join the eye movements to the next focused point in the screen. After the calibration step, the eye-tracker can be used on-line.



(a) (b)

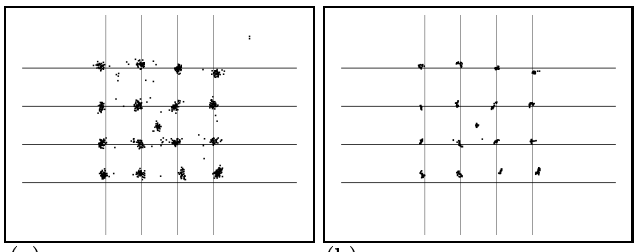
Figure 4: Off-line calibration step: eye-tracker measurements (a) corresponding to the calibration grid, (b) focused by the operator.

3.2 Gazed coordinates filtering

The gazed points measured by the eye-tracker are sensitive to noise, and sometimes unstable and inaccurate. The noise seems mainly due to the flicker of the eyelids and to the lighting variations. The unstability of the focused coordinates is essentially induced by unremitting and unconscious eye micro-movements. It means that images are never stable on the retina. The inaccuracy is the result of the eye-tracker measurement errors. The further the TV screen reproducing the camera picture from the operator, the great this error.

To stabilize and smooth the eye-tracker data and to differentiate focused points from outliers, a sliding averaging filter has been introduced. The number n of points, which are used to compute the mean value, is directly in relation with the minimum of measurements for which we consider that a point is really fixed. In practice, we set $n = 4$. So, the mean value of n successive eye-tracker coordinates gives the position of the focused point in the picture. The obtained variance σ^2 permits us to discriminate focused points from outliers. The gazed point is valid if the n successive coordinates extracted by the eye-tracker are in a m pixels wide square window ($\sigma < m$). In practice we set $m = 10$. In the other case, we consider that the point is not significant.

In order to estimate the eye-tracker data accuracy, we have asked the operator to focus on each of the 16 intersection points of the calibration grid, during approximatively 2 seconds. Figure 5.a presents the focused pixel coordinates (X_p, Y_p) resulting from this experiment. Figure 5.b, which presents the focused filtered points, permits us to outline the efficiency of the filtering process. All isolated points, assumed meaning less, are suppressed. The stacks of points corresponding to the focused regions are still present, but combined and purified.



(a) (b)

Figure 5: Influence of the meaning-sliding filter: (a) focused points extracted by the eye-tracker, (b) focused filtered points.

4 Visual Servoing

The *image-based visual servoing* consists in specifying a task as the regulation in the image of a set of visual features [4] [6]. Let us denote $\underline{s}$ the set of focused coordinates in the image used in the visual servoing task. To elaborate a control law based on $\underline{s}$, we need to know the interaction matrix $L_{\underline{s}}^T$, also called image Jacobian, defined by the classical equation [4]:

$$\dot{\underline{s}} = L_{\underline{s}}^T \, T_c \qquad (1)$$

where $\dot{\underline{s}}$ is the time variation of $\underline{s}$ due to the camera controlled velocities T_c.

A vision-based task $\underline{e}$ can thus be defined by:

$$\underline{e} = C(\underline{s} - \underline{s}_d) \quad (2)$$

where $\underline{s}_d$ is the desired value of the focused image coordinates, $\underline{s}$ is their current value (measured from the eye-tracker at each iteration of the control law), and where C, called combination matrix, has to be chosen such that $CL_{\underline{s}}^T$ is full rank. It can be defined as:

$$C = WL_{\underline{s}}^{T^+} \quad (3)$$

where W is a full rank matrix such that Ker W = Ker $L_{\underline{s}}^T$.

In order that $\underline{e}$ exponentially decreases, and then behaves like a first order decoupled system, the desired evolution of $\underline{e}$ takes the form [4]:

$$T_c = -\lambda \underline{e} \quad (4)$$

where T_c is the joint velocity given as input to the robot controller. In (4) λ is a proportional coefficient involved in the exponential convergence of $\underline{e}$, which has to be tuned in order to optimize the time to convergence of $\underline{e}$.

Positioning with respect to the gazed point

The task here consists in orientating the camera such that the gazed point of coordinates (X_p, Y_p) in pixels appears in the middle in the image (of coordinates (X_c, Y_c)). In order to obtain metrix units, X_p and Y_p are transformed by the relation:

$$X = (X_p - X_c) \cdot \frac{m_x}{f} \qquad Y = (Y_p - Y_c) \cdot \frac{m_y}{f} \quad (5)$$

where f is the current value of the focal length, and m_x, m_y the size of an elementary pixel of the CCD matrix. As well known in visual servoing, the convergence, stability and robustness of closed-loop control laws are not disturbed if these values are not accurately determined.

Using [7], we obtain $\underline{s} = (X, Y)$ and $\underline{s}_d = (0, 0)$. Since only pan and tilt are controlled, the interaction matrix $L_{\underline{s}}^T$ is given by:

$$L_{\underline{s}}^T = \begin{pmatrix} XY & -(1+X^2) \\ 1+Y^2 & -XY \end{pmatrix} \quad (6)$$

In this case, we have $W = \mathbb{I}_2$ and thus $C = L_{\underline{s}}^{T^{-1}}$. The control law (4) can finally be detailed as follow:

$$T_c = \begin{pmatrix} \Omega_x \\ \Omega_y \end{pmatrix} = \begin{pmatrix} -\lambda \cdot \frac{Y}{1+X^2+Y^2} \\ -\lambda \cdot \frac{-X}{1+X^2+Y^2} \end{pmatrix} \quad (7)$$

In Figure 6, the structure of the proposed visual servoing scheme is shown.

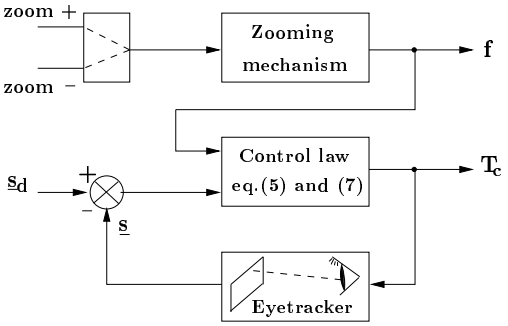


Figure 6: Visual servoing controller.

5 Experimental results

In this section, experimental results are presented in order to validate the proposed scheme.

5.1 Focusing task

In this experiment (see Figure 7), we have asked the operator to look continuously at the letter "s" in the word "robots". This letter takes position on the right of the image captured by the camera and projected on the TV screen in front of the operator equipped with the eye-tracker (see Figure 7.a). The gaze control is then activated in order to make this letter appear at the center of the image.

Figures 7.a, 7.b, 7.c respectively present the images seen by the pilot at the beginning, the middle and at the end of the experiment.

The corresponding image errors $(\underline{s} - \underline{s}_d)$ and pan and tilt velocities versus the number of sampling periods are presented in Figure 8.a and 8.b respectively. The results obtained emphasize the exponential convergence of the task function.

5.2 Target tracking task

In this experiment, similar to a parking survey (see Figure 9), the goal is to track a person appearing in the field of view (see Figure 9.b) and to gaze more precisely on him by using the zooming mechanism.

Figures 9.a, 9.b and 9.c show the images viewed by the pilot at the beginning, the middle and after 20 seconds respectively. In these images, the white cursor indicates the region focused on in the image by the pilot.

In Figure 10.a, all instantaneous coordinates focused on in the screen and computed by the eye-tracker during the experience are shown. The corresponding filtered points, which are used in the control law, are presented in Figure 10.b. The errors $(\underline{s} - \underline{s}_d)$ and camera velocities are presented in Figure 11.b and 11.c respectively.

This experience can be divided into three phases:

Figure 7: Gazing on the "s" letter: (a) first image, (b) intermediate image, (c) last image.

- a **waiting stage** where the operator gazes on the middle of the screen (correspond to the 600 first iterations in Figures 11.b and 11.c.);
- a **detection and convergence step** (from iteration 600 to 700 in Figures 11.b and 11.c.), where the operator detects the apparition of the person in the left of the screen (see Figure 9.b) and where the control law brings the image of the pedestrian in the middle of the screen. Focused points corresponding to this stage are visible in the left half of Figure 10.a;
- a **tracking stage** where the control law keeps the person around the middle of the screen, despite its motion. The pedestrian's motion corresponds to an horizontal translation in the image. Thus, error $y - y_d$ and tilt velocity are very small. The accumulation of points in the middle of Figure 10.b underlines this tracking step.

In spite of the focal length variations (see Figure 11.a) and of the person's displacements, the tracking and survey tasks are achieved by keeping the pedestrian approximatively in the middle of the image. The accumulation of points in the middle of the Figure 10.b illustrates the stability and the robustness of the control law.

6 Discussion and future work

A solution was proposed to pilot a camera mounted on the end effector of a robot using eye movements. A closed-loop visual servoing control scheme, which performs the regulation of the selected vision-based task was also presented.

The whole scheme has been implemented and tested on an experimental testbed. On various experiments (focusing and target tracking with several pilots), the validity and stability of our approach has been demonstrated.

The introduction of a coder-decoder (Co-Dec) in the data processing will be considered in the future. The camera frame will be compressed, broadcasted along a 256 k-bits RNIS channel and restored, after decoding, to the TV screen in front of the pilot. In that case, the delay induced by the coding and decoding steps will have to be taken into account. This work will make possible human-controlled survey applications such as long range inspection in sensitive sites.

References

[1] P.K. Allen, A. Timcenko, B. Yoshimi, and P. Michelman. Automated tracking and grasping of a moving object with a robotic hand-eye system. *IEEE Trans. on Robotics and Automation*, 9(2):152–165, April 1993.

[2] A. Castaño and S. Hutchinson. Visual compliance: Task-directed visual servo control. *IEEE Trans. on Robotics and Automation*, 10(3):334–342, June 1994.

[3] B. Espiau. Asservissement visuel et commande de la distance focale. Technical report, INRIA, July 1995.

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

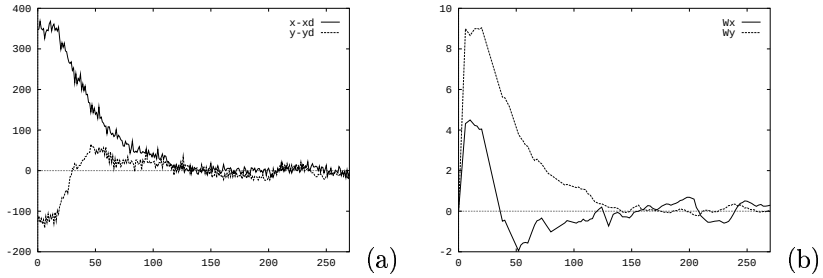
[5] G. Hager and P. Belhumeur. Real-time tracking of image regions with changes in geometry and illumination. In *IEEE Int. Conf. on Computer Vision and Pattern Recognition*, pages 403–410, San Francisco, USA, June 1996.

[6] K. Hashimoto Editor. *Visual Servoing : Real Time Control of Robot manipulators based on visual sensory feedback.* World Scientific Series in Robotics and Automated Systems, Vol 7, World Scientific Press, Singapore, 1993.

[7] K. Hosoda, H. Moriyama, and M. Asada. Visual servoing utilizing zoom mechanism. In *IEEE Int. Conf. on Robotics and Automation*, Nagoya, Japan, May 1995.

[8] N.P. Papanikolopoulos, P.K. Khosla, and T. Kanade. Visual tracking of a moving target by a camera mounted on a robot: A combination of control and vision. *IEEE Trans. on Robotics and Automation*, 9(1):14–35, February 1993.

[9] L.E. Weiss, A.C. Sanderson, and C.P. Neuman. Dynamic sensor-based control of robots with visual feedback. *IEEE Journal of Robotics and Automation*, 3(5):404–417, October 1987.

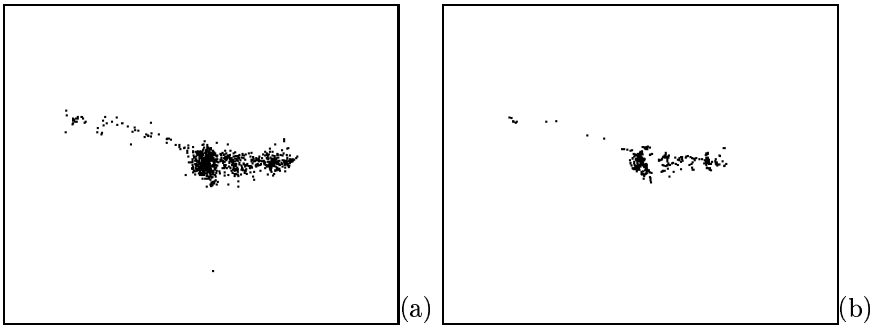


(a) (b)

Figure 8: Gazing on the "s" letter: (a) error $(\underline{s} - \underline{s}_d)$ in pixels, (b) velocities T_c in dg/s.

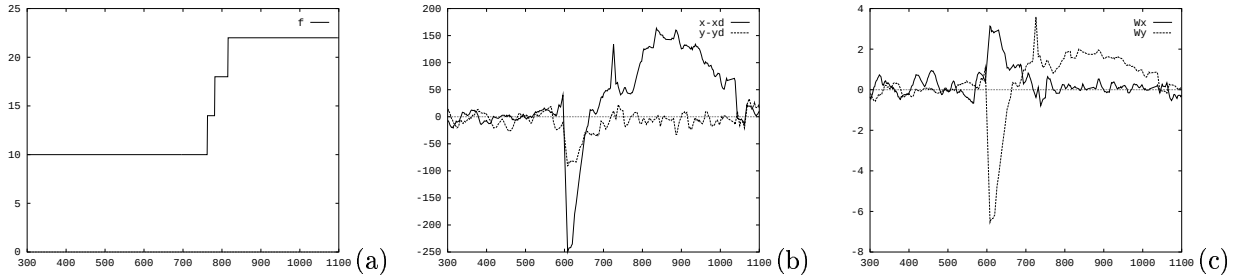
(a) (b) (c)

Figure 9: Target tracking: (a) first image, (b) intermediate image near iteration 600, (c) last image.



(a) (b)

Figure 10: Target tracking: (a) gazed coordinates extracted by the eye-tracker, (b) focused filtered coordinates $\underline{s}$.



(a) (b) (c)

Figure 11: Target tracking: (a) focal length in mm, (b) error $(\underline{s} - \underline{s}_d)$ in pixels, (c) velocities T_c in dg/s.