In Proceedings of the 8th ESA Workshop on Advanced Space Technologies for Robotics and Automation
'ASTRA 2004' ESTEC, Noordwijk, The Netherlands, November 2 - 4, 2004

Recent results in visual servoing for robotics applications

François Chaumette, Eric Marchand
IRISA / INRIA Rennes
Campus de Beaulieu, 35 042 Rennes-cedex, France
E-Mail : Firstname.Lastname@irisa.fr

Abstract

This paper presents new advances in the £eld of visual servoing. More precisely, we consider the case where complex objects are observed by a camera. In a £rst part, planar objects of unknown shape are considered using image moments as input of the image-based control law. In the second part, a pose estimation and tracking algorithm is described to deal with real objects whose 3D model is known. For each case, experimental results obtained with an eye-in-hand system are presented.

1 Introduction

Visual servoing techniques consist in using the data provided by one or several cameras in order to control the motions of a robotic system [7, 8]. A large variety of positionning tasks, or mobile target tracking, can be implemented by controling from one to all the n degrees of freedom of the system. Whatever the sensor con£guration, which can vary from one on-board camera on the robot end-effector to several free-standing cameras, a set of k measurements as to be selected at best, allowing to control the m degrees of freedom desired. A control law has also to be designed so that these mesurements $\mathbf{s}(t)$ reach a desired value $\mathbf{s}^*$, de£ning a correct realization of the task. A desired trajectory $\mathbf{s}^*(t)$ can also be tracked. The control principle is thus to regulate to zero the error vector $\mathbf{s}(t)-\mathbf{s}^*(t)$. With a vision sensor providing 2D measurements, potential visual features are numerous, since as well 2D data such as coordinates of feature points in the image can be considered, as 3D data provided by a localization algorithm exploiting the extracted 2D features (see Figure 1). It is also possible to combine 2D and 3D visual features to take the advantages of each approach while avoiding their respectives drawbacks [11]

In this paper, we present recent results in visual servoing for positioning tasks with respect to complex objects. In the next section, we recall some modeling aspects. In Section 3, a pose estimation and tracking algorithm is described to deal with real objects whose 3D model is known. In that case, any visual servoing scheme can be used: image-based (2D), position-based (3D), or hybrid scheme (2 1/2D). Finally, experimental results using image motion estimation are presented in Section 4.

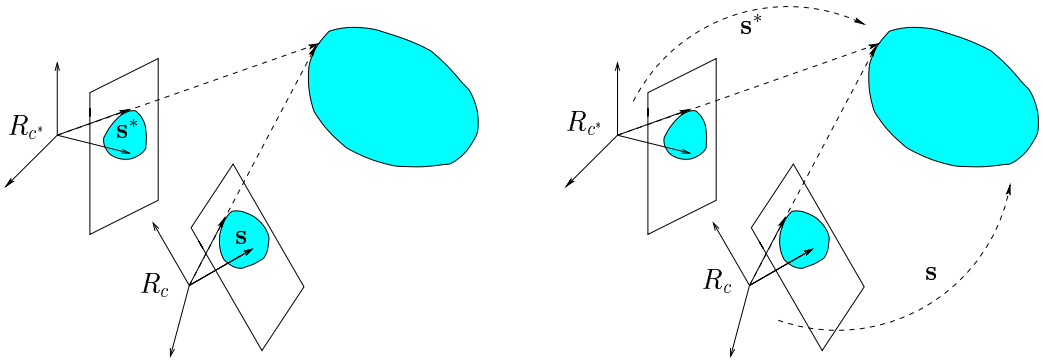

Figure 1: 2D or 3D visual servoing: to bring the camera frame from R_c to R_{c^*}, visual features directly extracted from the image are used in 2D visual servoing (left), while in 3D visual servoing, features estimated through a pose estimation or a 3D reconstruction are considered (right).

2 Visual features modeling

A set $\mathbf{s}$ of k visual features can be taken into account in a visual servoing scheme from the moment it can be written:

$$\mathbf{s} = \mathbf{s}(\mathbf{r}(t)) \tag{1}$$

where $\mathbf{r}(t)$ describes the pose at the instant t between the camera frame and the target frame. The variation of $\mathbf{s}$ can be linked to the relative kinematic motion $\mathbf{v}$ between the camera and the scene.

$$\dot{\mathbf{s}} = \frac{\partial \mathbf{s}}{\partial \mathbf{r}}\, \dot{\mathbf{r}} = \mathbf{L}_{\mathbf{s}}\, \mathbf{v} \tag{2}$$

where $\mathbf{L}_{\mathbf{s}}$ is the interaction matrix related to $\mathbf{s}$.

Case of an eye-in-hand system: If we consider a camera mounted on a robot end-effector, we obtain:

$$\dot{\mathbf{s}} = \mathbf{L}_{\mathbf{s}}^{T}\, {}^{c}\mathbf{W}_{n}\, {}^{n}\mathbf{J}_{n}(\mathbf{q})\, \dot{\mathbf{q}} + \frac{\partial \mathbf{s}}{\partial t} \tag{3}$$

where:

- ${}^{n}\mathbf{J}_{n}(\mathbf{q})$ is the robot Jacobian expressed in the frame R_n of its end-effector ;
- $\frac{\partial \mathbf{s}}{\partial t}$ is the variation of $\mathbf{s}$ due to the potential object motion (generally unknown) ;
- ${}^{c}\mathbf{W}_{n}$ is the transformation of the kinematic screw to go from its expression in the camera frame R_c to the frame R_n. This transformation matrix can be estimated quite precisely by using an eye-to-hand calibration technique. However, visual servoing is in general robust enough to admit large modeling errors as well in this transformation matrix as in the robot Jacobian, as the camera intrinsic parameters [11].

Case of an eye-to-hand system: In a similar way, if we consider a free-standing camera observing the end-effector of a robot arm, the variation of the visual features rigidly linked with this end-effector are given by:

$$\dot{\mathbf{s}} = -\mathbf{L}_{\mathbf{s}}\, {}^{c}\mathbf{W}_{n}\, {}^{n}\mathbf{J}_{n}(\mathbf{q})\, \dot{\mathbf{q}} + \frac{\partial \mathbf{s}}{\partial t} \tag{4}$$

where $\frac{\partial \mathbf{s}}{\partial t}$ now describes the variations of $\mathbf{s}$ due to a potential motion of the free-standing camera. We can note the difference of sign between the equations (3) and (4). This difference becomes naturally clear with the change of sensor con£guration with respect to the control variables (see Figure 2).

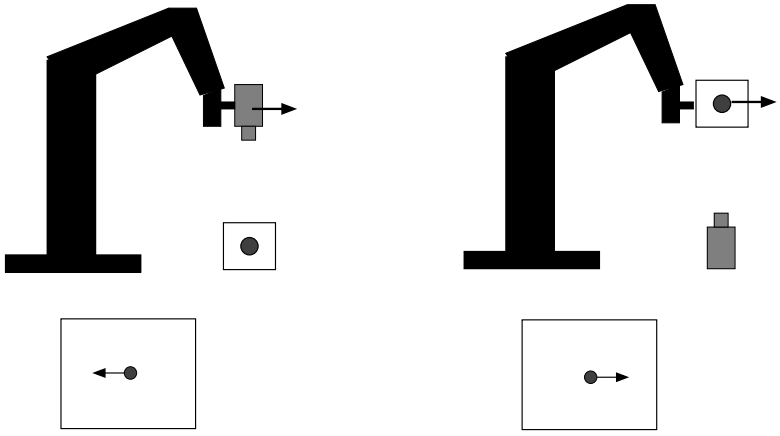

Figure 2: Difference of con£guration (at the top) and of the effect produced in the image acquired by the camera (at the bottom)

In each case (eye-in-hand or eye-to-hand), the interaction matrix plays an essential role. Indeed, if we consider for instance an eye-in-hand system and the camera velocity as input of the robot controller, we obtain when the control law is designed to try to obtain an exponential decoupled decrease of the error:

$$\mathbf{v}_c = -\lambda \widehat{\mathbf{L}_{\mathbf{s}}}^{+} (\mathbf{s} - \mathbf{s}^{*}) - \widehat{\mathbf{L}_{\mathbf{s}}}^{+} \widehat{\frac{\partial \mathbf{s}}{\partial t}} \tag{5}$$

where λ is a proportional gain that has to be tuned to minimize the time-to-convergrence, $\widehat{\mathbf{L_s}}^+$ is the pseudo-inverse of a model or an approximation of the interaction matrix, and $\widehat{\frac{\partial \mathbf{s}}{\partial t}}$ an estimation of the target velocity. The analytical form of the interaction matrix has been determined for many possible visual features, such as image point coordinates, 2D straight lines, 2D ellipses, image moments, 3D coordinates of points, etc. From the selected visual features, the behavior of the system will have particular properties as for stability, robustness with respect to noise or to calibration errors, robot 3D trajectory, etc. It is thus extremely important to choose adequate visual features for each robot task or application. Promising results have been obtained recently using image moments [13]. The £rst interest of using image moments is that they provide a generic and geometrically intuitive representation of any object, with simple or complex shapes that can be segmented in an image. They can also be extracted from a set of image points tracked along an image sequence by simple summation of polynomials that depend on the points position.

Furthermore, as already noticed, an important aspect is to determine the visual features to use in the control scheme in order to obtain an optimal behavior of the system. A good objective is to design a decoupled control scheme, i.e. to try to associate each robot degree of freedom with only one visual feature through a linear relation. A such totally decoupled and linear control would be ideal. Currently, it is possible to decouple the translational motions from the rotational ones. This decoupled control can be obtained using moment invariants as fully described in [13]. In few words, a set of adequate combination of moments has been selected so that the related interaction matrix $\mathbf{L_s}$ is as near as possible of a triangular constant matrix.

Experimental results are reported on Figure 3. They have beeen obtained with a six degrees of freedom eye-in-hand robot. The goal was to position the camera so that the corresponding image is the same as one image acquired during an off line learning step. Several points of interest have been extracted using the Harris detector and tracked using a SSD algorithm [14]. Image moments have then be computed from the coordinates of these points. The plots depicted on Figure 3 show that the system converges with an exponential decrease.

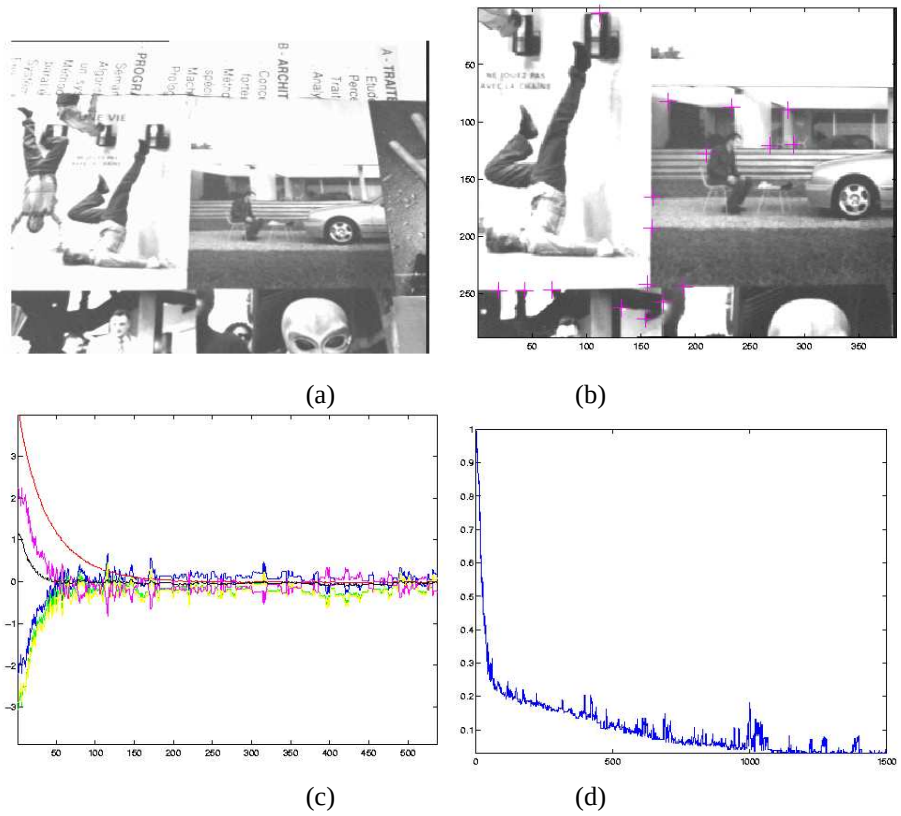

Figure 3: Results for complex images: (a) initial image, (b) desired image, (c) robot velocities versus time, (d) visual features errors mean versus time

3 Robust model-based tracking

This section addresses the problem of realizing visual servoing tasks by using complex objects in real environments. For that, we present a real-time 3D model-based tracking of objects in monocular image sequences. This fundamental vision problem has applications in many domains ranging from augmented reality to visual servoing, and even medical imaging or industrial applications. The main advantage of a 3D model-based method is that the knowledge about the scene (the implicit 3D information) allows improvements of robustness and performance by being able to predict hidden movement of the object and acts to reduce the effects of outlier data introduced in the tracking process.

In the related literature, geometric primitives considered for the estimation are often points[5], segments, contours or points on the contours, conics, cylindrical objects, or a combination of these different features. Another important issue is the registration problem. Purely geometric, or numerical and iterative approaches may be considered. Linear approaches use a least-squares method to estimate the pose. Full-scale non-linear optimization techniques ([10, 6, 9]) consists of minimizing the error between the observation and the forward-projection of the 3D model. In this case, minimization is handled using numerical iterative algorithms such as Newton-Raphson or Levenberg-Marquardt. The main advantage of these approaches are their accuracy. The main drawback is that they may be subject to local minima and, worse, divergence.

Our method is fully described in [4]. Pose computation is formulated in terms of a full scale non-linear optimization using a 2D virtual visual servoing scheme [12, 15]. Our method takes the 2D visual servoing framework by controlling the motion of a virtual camera so that the projection in the image of the object model perfectly £ts with the current position of the object in the image acquired by the real camera. We thus obtain an image feature-based system which is capable of treating complex scenes in real-time without the need for markers. Contributions can be exhibited at three different levels:

- as already explained in the previous section, the analytical form of the interaction matrices $\mathbf{L_s}$ related to complex visual features including ellipses, cylinders, points, distances and any combination of these is easily obtained. Determining an accurate approximation of this matrix is essential to obtain the convergence of the visual servoing. In [4], a complete derivation of interaction matrices for distances to lines, ellipses and cylinders are given.

- the widely accepted statistical techniques of robust M-estimation are employed. This is introduced directly in the virtual visual servoing control law by weighting the con£dence on each feature. The Median Absolute Deviation (MAD) is used as an estimate of the standard deviation of the inlier data. Statistically robust pose computation algorithm, suitable for real-time tracking techniques, have been considered.

- the formulation for tracking objects is dependent on correspondences between local features in the image and the object model. In an image stream, these correspondences are given by the local tracking of features in the image. In our method, low level tracking of the contours is implemented via an adequate algorithm, called Moving Edges algorithm [1]. A local approach such as this is ideally suited to real-time tracking due to an ef£cient 1D search normal to a contour in the image. In a 'real world' scenario, some features may be incorrectly tracked, due to occlusion, changes in illumination and miss-tracking. Since many point-to-curve correspondences are made, the method given here has many redundant features which favors the use of robust statistics.

Any visual servoing control law can be used using the output of our tracker: image-based (2D), position-based (3D) or hybrid scheme (2 1/2D). In the presented experiments, we have considered the 2 1/2D approach [11]. It consists in combining visual features obtained directly from the image, and features expressed in the Euclidean space. The 3D information can be retrieved either by a projective reconstruction obtained from the current and desired images, either by a pose estimation algorithm. In our context, since the pose is an output of our tracker, we consider in this paper the latter solution.

The complete implementation of the robust visual servoing task, including tracking and control, was carried out on an experimental test-bed involving a CCD camera mounted on the end effector of a six d.o.f robot. Images were acquired and processed at video rate (50Hz). In such experiments, the image processing is potentially very complex. Indeed extracting and tracking reliable points in real environment is a non trivial issue. The use of more complex features such as the distance to the projection of 3D circles, lines, and cylinders has been demonstrated in [4] in an augmented reality context. In all experiments, the distances are computed using the Moving Edges algorithm previously described. Tracking is always performed at below frame rate (usually in less than 10ms).

In all the £gures depicted, current position of the tracked object appears in green while its desired position appears in blue. Three objects where considered: a micro-controller (Figure 4), an industrial emergency switch (Figure 5) and a video multiplexer (Figure 6). To validate the robustness of the algorithm, the objects were placed in a highly textured environment as shown in Fig. 4, 5 and 6. Tracking and positioning tasks were correctly achieved. Multiple temporary

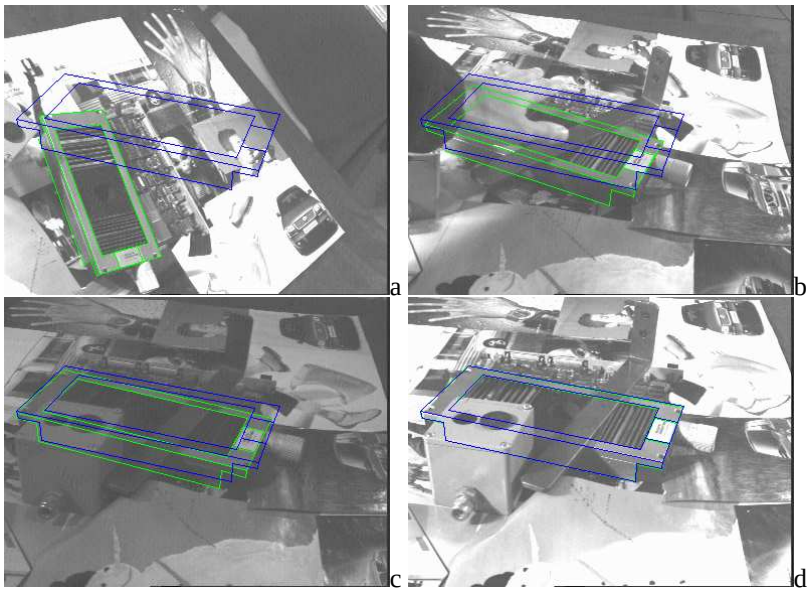

Figure 4: Tracking in complex environment within visual servoing: Images are acquired and processed at video rate (25Hz). Blue: desired position de£ned by the user. Green: position measured after pose calculation. (a) £rst image initialized by hand, (b) partial occlusion with hand, (c) lighting variation, (d) £nal image with various occlusions

and partial occlusions by an hand and various work-tools, as well as modi£cation of the lighting conditions were imposed during the realization of the positioning task. On the third experiments (see Figure 6), after a complex positioning task (note that some object faces appeared while other disappeared), the object is handled by hand and moved around. Since the visual servoing task has not been stopped, robot is still moving in order to maintain the rigid link between the camera and the object.

For the second experiment, plots are also shown which helps to analyse the pose estimation, the robot velocity and the error vector. We can see that the robot velocity reaches 23 cm/s in translation and 85 dg/s in rotation. In other words, less than 35 frames were acquired during the entire positioning task up until convergence despite the large displacement to achieve (see Figure 5e). Therefore the task was accomplished in less than 1 second. Let us note that in all these experiments, neither a Kalman £lter (or other prediction process) nor the camera displacement were used to help the tracking.

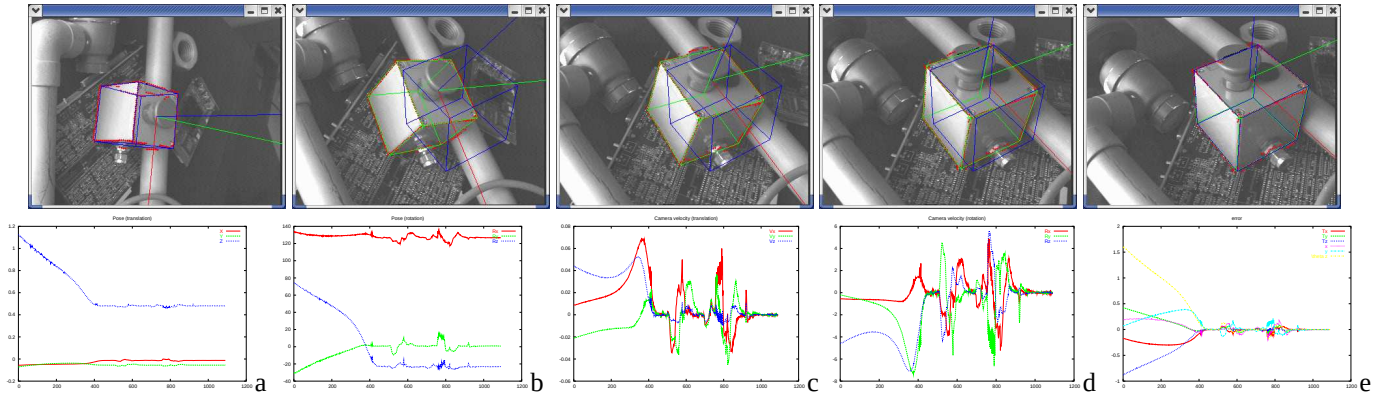

Figure 5: 2D 1/2 visual servoing experiments: on these £ve snapshots, the tracked object appears in green and its desired position in the image in blue. Plots correspond to (a) pose (translation), (b) pose (rotation), (c-d) camera velocity in rotation and translation, (e) error vector $\mathbf{s} - \mathbf{s}^*$

.

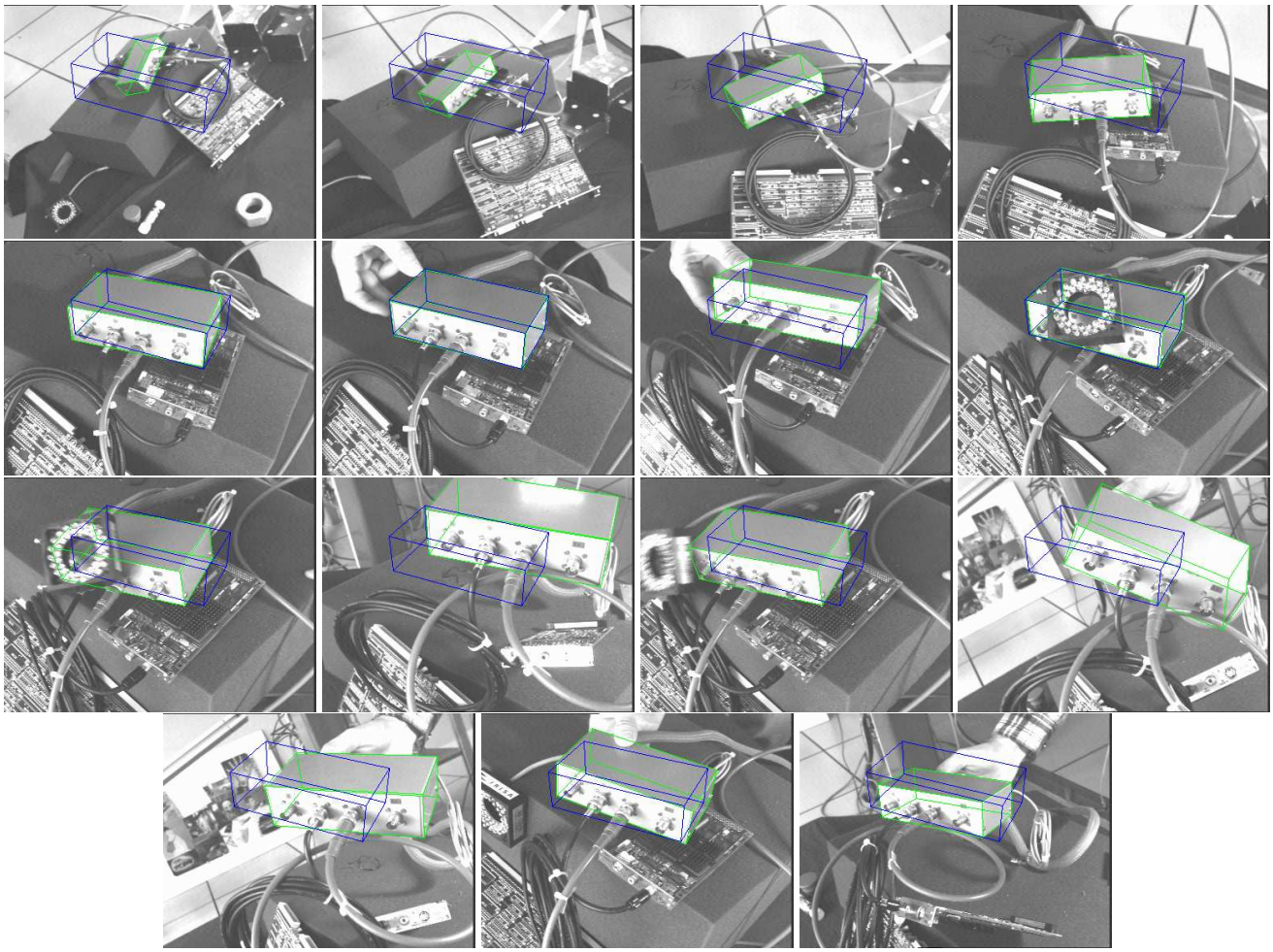

Figure 6: 2D 1/2 visual servoing experiments: on these snapshots the tracked object appears in green and its desired position in the image in blue. The six £rst images have been acquired during an initial visual servoing step where the object is motionless. In the reminder images, object is moving along with the robot.

4 Image motion in visual servoing

To end this paper, we present some experimental results obtained on complex environments using an image motion estimation between two successive images. The task that corresponds to the images of Figure 7 consists in controlling the pan and tilt of a camera so that a moving pedestrian always remains in the camera £eld of view whatever his motion. We can note the robustness of the image processing and of the control law with respect to non rigid motion. More details are given in [3], as well as other experiments obtained for submarine robotics applications.

References

[1] P. Bouthemy. A maximum likelihood framework for determining moving edges. *IEEE Trans. on Pattern Analysis and Machine intelligence*, 11(5):499–511, May 1989.

[2] F. Chaumette. Potential problems of stability and convergence in image-based and position-based visual servoing, *The Con¤uence of Vision and Control*, pp. 66-78, *LNCIS Series*, No 237, Springer-Verlag, 1998.

[3] A. Crétual, F. Chaumette. Application of motion-based visual servoin to target tracking. *Int. Journal of Robotics Research*, 20(11), November 2001.

[4] A. Comport, E. Marchand, and F. Chaumette. A real-time tracker for markerless augmented reality. *ACM/IEEE Int. Symp. on Mixed and Augmented Reality, ISMAR'03*, pp. 36–45, Tokyo, Japan, October 2003.

[5] D. Dementhon and L. Davis. Model-based object pose in 25 lines of codes. *Int. Journal of Computer Vision*, 15:123–141, 1995.

[6] T. Drummond and R. Cipolla. Real-time visual tracking of complex structures. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 27(7):932–946, July 2002.

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

[8] S. Hutchinson, G. Hager, P. Corke: A tutorial on visual servo control, *IEEE Trans. on Robotics and Automation*, 12(5):651-670, October 1996.

[9] D. Kragic and H.I. Christensen. Con¤uence of parameters in model based tracking. In *IEEE Int. Conf. on Robotics and Automation, ICRA'03*, volume 4, pages 3485–3490, Taipei, Taiwan, September 2003.

[10] D. Lowe. Robust model-based motion tracking trough the integration of search and estimation. *Int. Journal of Computer Vision*, 8(2):113–122, 1992.

[11] E. Malis, F. Chaumette, and S. Boudet. 2 1/2 D visual servoing. *IEEE Trans. on Robotics and Automation*, 15(2):238–250, April 1999.

[12] E. Marchand and F. Chaumette. Virtual visual servoing: a framework for real-time augmented reality. *Eurographics'02 Conf. Proc.*, Vol. 21(3) of *Computer Graphics Forum*, pp. 289–298, Saarebrücken, Germany, September 2002.

[13] O. Tahri, F. Chaumette. Application of moment invariants to visual servoing *IEEE Int. Conf. on Robotics and Automation, ICRA'03*, Vol. 3, pp. 4276-4281, Taipei, Taiwan, September 2003.

[14] J. Shi and C. Tomasi. Good features to track. In *IEEE Int. Conf. on Computer Vision and Pattern Recognition, CVPR'94*, pages 593–600, Seattle, June 1994.

[15] V. Sundareswaran and R. Behringer. Visual servoing-based augmented reality. In *IEEE Int. Workshop on Augmented Reality*, San Francisco, November 1998.

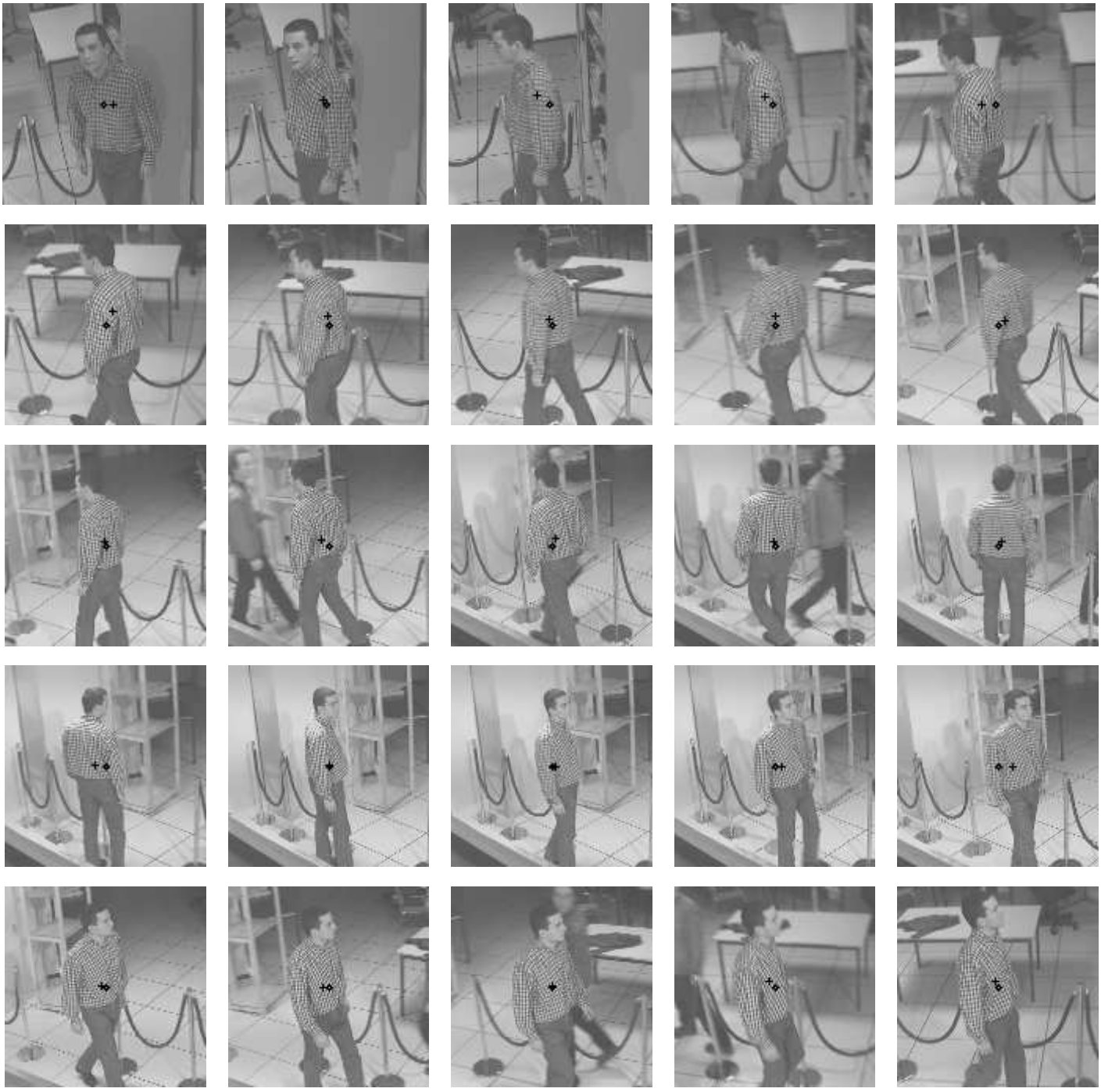

Figure 7: Camera pan/tilt control for a tracking task.