# Contour-based 3D motion recovery while zooming

E. Martínez Marroquín[a] * and C. Torras Genís[b†]

[a]Communications and Signal Theory Department, Enginyeria La Salle,
Universitat Ramon Llull, Pge. Bonanova 8, 08022 Barcelona, Spain

[b]Institut de Robòtica i Informàtica Industrial, CSIC-UPC,
Llorens i Artigas 4-6 , 08028 Barcelona, Spain

This paper considers the problem of 3D motion recovery from a sequence of monocular images while zooming. Unlike the common trend based on point matches, the proposed method relies on the deformation of an active contour fitted to a reference object. We derive the relation between the contour deformation and the 3D motion components, assuming time-varying focal length and principal point. This relation allows us to present a method to extract the rotation matrix and the scaled translation along the optical axis.

*Keywords: 3D motion recovery, egomotion, visual odometry, autocalibration, time-varying internal parameters, active contours, planar constraints.*

## 1. INTRODUCTION

The ability to zoom provides an image definition that eases a range of visual tasks common in robot vision, such as structure recovery or recognition. However, camera zooming invalidates most of the current solutions to computer vision problems (e.g., tracking or calibration), which assume constant intrinsic camera parameters, and therefore demands new approaches [1,2]. Zooming does not only change the focal length but also the principal point, due to optical and mechanical misalignments in the lens system of the camera [3,4]. The rest of intrinsic camera parameters (e.g., pixel size and aspect ratio) remain constant for long periods of time [5] and may be assumed known.

The process of calibration with the aid of a calibration pattern [6,7] is inapplicable in real time or in cases where the camera optical parameters undergo frequent changes. Different approaches have recently emerged for autocalibrating the camera assuming time-varying internal parameters [4,8,9]. They are based only on point matches and do not exploit the constraints on the geometric structure of the scene. Starting in [10], efforts has been devoted to incorporate geometric constraints in the reconstruction process [11–14]. The present work is based on an active contour and explicitly takes into account the particular geometry of a planar structure.

It is known that the 3D structure and motion can be recovered from a sequence of images [15,16]. This requires a measure of the visual motion on the image plane and a model that relates this motion to the real 3D motion. The bottleneck when trying to bring this into practice is the computation of visual motion, which requires at least a set of feature matches between frames. Moreover, common methods for feature matching perform particularly poorly when zooming. Noting that the cumulative research on active contours [17–19] provides an efficient tracking of objects, this work has been motivated by the idea of building an algorithm for 3D motion recovery upon an active contour tracker.

Previous works by the authors highlight the

*Corresponding author. Tel: +34 932902440; fax: +34 932902416.
E-mail addresses: elisa@salleurl.edu, ctorras@iri.upc.es

†This work has been partially supported by the research grant "Navegación autónoma de robots guiados por objetivos visuales"' CICYT DPI2000-1352-C02-01 of the Spanish Science and Technology Council

feasibility of recovering 3D structure and motion from the analysis of an active contour fitted to a reference object. This is shown for different degrees of camera calibration [20,21] and for uncalibrated cameras with constant intrinsic parameters [22,23]. Here we extend the analysis to the case of time-varying internal calibration parameters due to zooming.

The work described in this paper stems from a project aimed at the development of a walking robot for exploratory tasks [24]. Part of this project is concerned with the design of a visual system to provide the robot with enough autonomy to reach a visual target in natural scenes. The paper is organized as follows. Section 2 relates the deformation of a contour to the 3D motion components and the internal calibration parameters. Then, Section 3 describes the process followed to recover the 3D motion components. Section 4 shows two examples of the experiments conducted to test the method. Finally, we draw some conclusions in Section 5.

## 2. TWO-VIEW GEOMETRY OF A PLANAR CONTOUR

An active contour is fitted to the occluding contour $\mathbf{D_0}(s)$ of a reference object, which is marked on-line by the operator and may have any shape. This occluding contour can be written in parametric form as $\mathbf{D_0}(s) = (X_0(s), Y_0(s), Z_0(s))^T$ where $s$ is a parameter that increases as the contour is traversed.

When there is a relative motion between the camera and the object, the reference object presents a new occluding contour which we denote $\mathbf{D}(s)$. Under weak perspective conditions, i.e. when the object fits in a small field of view and the depth variation of its points is small compared to their distances to the camera, the occluding contour of the object can be assumed to be a 3D curve that moves rigidly in 3D space. As we are interested in tracking a distant target, both conditions hold. Therefore,

$$\mathbf{D}(s) = \mathbf{R}\mathbf{D_0}(s) + \mathbf{T}, \tag{1}$$

where $\mathbf{R}$ is the rotation matrix and $\mathbf{T}$ is the translation vector corresponding to the 3D rigid motion.

Moreover, the weak perspective conditions allow us to assume also a simplified camera model to analyse the projection of the 3D curve onto the image plane.

The projection $\mathbf{d_0}(s)$ (called, hereafter, the template) of the 3D curve in the initial frame, $\mathbf{D_0}(s)$, is

$$\mathbf{d_0}(s) = \frac{f^{(0)}}{Z_0}\begin{bmatrix} K_u & 0 \\ 0 & K_v \end{bmatrix}\begin{bmatrix} X_0(s) \\ Y_0(s) \end{bmatrix} + \begin{bmatrix} u_0^{(0)} \\ v_0^{(0)} \end{bmatrix}, \tag{2}$$

where $f^{(i)}, u_0^{(i)}, v_0^{(i)}$ are the focal length and principal point for frame $i$; $K_u, K_v$ denote the pixel size, and $Z_0$ is the distance from the camera to the target at the reference frame.

The projection of the 3D curve in a subsequent frame $i$ is

$$\begin{gathered}\mathbf{d}(s) = \frac{f^{(i)}}{Z_0 + T_z}\begin{bmatrix} K_u & 0 \\ 0 & K_v \end{bmatrix} \times \\ \left(\begin{bmatrix} R_{11} & R_{12} \\ R_{21} & R_{22} \end{bmatrix}\begin{bmatrix} X_0(s) \\ Y_0(s) \end{bmatrix} + \begin{bmatrix} T_x \\ T_y \end{bmatrix}\right) + \begin{bmatrix} u_0^{(i)} \\ v_0^{(i)} \end{bmatrix},\end{gathered} \tag{3}$$

where $R_{ij}$ are the elements of the rotation matrix and $T_i$ are the elements of the translation vector.

The geometry that relates a view $i$ of the planar contour with the template is derived by combining equations (2) and (3),

$$\begin{aligned}&\mathbf{d}(s) - \begin{bmatrix} u_0^{(i)} \\ v_0^{(i)} \end{bmatrix} = \\ &\frac{f^{(i)}}{f^{(0)}}\frac{Z_0}{Z_0 + T_z}\begin{bmatrix} K_u & 0 \\ 0 & K_v \end{bmatrix}\begin{bmatrix} R_{11} & R_{12} \\ R_{21} & R_{22} \end{bmatrix}\begin{bmatrix} \frac{1}{K_u} & 0 \\ 0 & \frac{1}{K_v} \end{bmatrix} \times \\ &\left(\mathbf{d_0}(s) - \begin{bmatrix} u_0^{(0)} \\ v_0^{(0)} \end{bmatrix}\right) + \\ &\frac{f^{(i)}}{Z_0 + T_z}\begin{bmatrix} K_u & 0 \\ 0 & K_v \end{bmatrix}\begin{bmatrix} T_x \\ T_y \end{bmatrix}.\end{aligned}$$

The above equation can be rewritten as

$$\mathbf{d}(s) - \begin{bmatrix} u_0^{(i)} \\ v_0^{(i)} \end{bmatrix} = \mathbf{L}\left(\mathbf{d_0}(s) - \begin{bmatrix} u_0^{(0)} \\ v_0^{(0)} \end{bmatrix}\right) + \mathbf{p},$$

where

$$\mathbf{L} = \frac{f^{(i)}}{f^{(0)}}\frac{Z_0}{Z_0 + T_z}\begin{bmatrix} R_{11} & R_{12}\frac{K_u}{K_v} \\ R_{21}\frac{K_v}{K_u} & R_{22} \end{bmatrix}$$

and

$$\mathbf{p}=\frac{f^{(i)}}{f^{(0)}}\frac{1}{Z_0+T_z}\begin{bmatrix}K_u & 0\\ 0 & K_v\end{bmatrix}\begin{bmatrix}T_x\\ T_y\end{bmatrix}. \tag{4}$$

The difference between the curve at a particular instant and the template is

$$\mathbf{d}(s)-\begin{bmatrix}u_0^{(i)}\\ v_0^{(i)}\end{bmatrix}-\mathbf{d_0}(s)+\begin{bmatrix}u_0^{(0)}\\ v_0^{(0)}\end{bmatrix}=$$
$$(\mathbf{L}-\mathbf{I})\left(\mathbf{d_0}(s)-\begin{bmatrix}u_0^{(0)}\\ v_0^{(0)}\end{bmatrix}\right)+\mathbf{p}, \tag{5}$$

where $\mathbf{I}$ is the $2\times 2$ identity matrix. This equation can be rewritten as

$$\mathbf{d}(s)-\mathbf{d_0}(s)=$$
$$(\mathbf{L}-\mathbf{I})\left(\mathbf{d_0}(s)-\begin{bmatrix}u_0^{(0)}\\ v_0^{(0)}\end{bmatrix}\right)+\mathbf{p}+\Delta\mathbf{u}, \tag{6}$$

where $\Delta\mathbf{u}\triangleq\begin{bmatrix}u_0^{(i)}-u_0^{(0)}\\ v_0^{(i)}-v_0^{(0)}\end{bmatrix}$. Without loss of generality, the center of the template is assumed to be equal to the principal point in the initial frame, then equation (6) can be rewritten in terms of $\mathbf{d_0'}(s)$ and $\mathbf{d}'(s)$, that is, the projected contours referred to the template's centroid, as

$$\mathbf{d}'(s)-\mathbf{d_0'}(s)=(\mathbf{L}-\mathbf{I})\mathbf{d_0'}(s)+\mathbf{p}+\Delta\mathbf{u}. \tag{7}$$

This result shows that the changes in the contour at each frame correspond to affine deformations of the template.

The affine parameters are $\mathbf{L}$ and $\mathbf{r}\triangleq\mathbf{p}+\Delta\mathbf{u}$. These are recovered from the shape of the contour at each frame using an active contour tracker [17, 19], based on a Kalman filter(see [22] for details).

The pixel size and, hence, the aspect ratio are constant along a sequence, and they are usually provided with the camera specifications. Assuming a known aspect ratio $\mathcal{A}=\frac{K_u}{K_v}$, the $\mathbf{L}$ matrix can be rewritten as

$$\mathbf{L}=\frac{f^{(i)}}{f^{(0)}}\frac{Z_0}{Z_0+T_z}\begin{bmatrix}R_{11} & R_{12}\mathcal{A}\\ R_{21}\frac{1}{\mathcal{A}} & R_{22}\end{bmatrix}.$$

Then, without loss of generality, $\mathcal{A}$ can be assumed equal to one, and a simplified matrix $\mathbf{L_s}$ can be computed

$$\mathbf{L_s}=\frac{f^{(i)}}{f^{(0)}}\frac{Z_0}{Z_0+T_z}\begin{bmatrix}R_{11} & R_{12}\\ R_{21} & R_{22}\end{bmatrix}. \tag{8}$$

## 3. EXTRACTION OF 3D MOTION PARAMETERS

In this section we derive the relation between the affine parameters described above and the 3D motion components: 3D rotation $\mathbf{R}$ and 3D translation $\mathbf{T}$. The rotation matrix can be written in terms of the Euler angles,

$$\mathbf{R}=\mathbf{R_z}(\phi)\mathbf{R_x}(\theta)\mathbf{R_z}(\psi) \tag{9}$$

where $\mathbf{R_z}(\psi)$ and $\mathbf{R_z}(\phi)$ are rotation matrices about the $Z$ axis and $\mathbf{R_x}(\theta)$ is a rotation matrix about the $X$ axis.

Using the Euler notation to represent the rotation matrix, equation (8) can be rewritten as

$$\mathbf{L_s}=\frac{f^{(i)}}{f^{(0)}}\frac{Z_0}{Z_0+T_z}\mathbf{R_z}|_\mathbf{2}(\phi)\mathbf{R_x}|_\mathbf{2}(\theta)\mathbf{R_z}|_\mathbf{2}(\psi)=$$
$$\frac{f^{(i)}}{f^{(0)}}\frac{Z_0}{Z_0+T_z}\mathbf{R_z}|_\mathbf{2}(\phi)\begin{bmatrix}1 & 0\\ 0 & cos\theta\end{bmatrix}\mathbf{R_z}|_\mathbf{2}(\psi), \tag{10}$$

where $\mathbf{R}|_\mathbf{2}$ denotes the $2\times 2$ submatrix of $\mathbf{R}$. Then,

$$\mathbf{L_s}\mathbf{L_s}^T=$$
$$\mathbf{R_z}|_\mathbf{2}(\phi)\begin{bmatrix}\left(\frac{f^{(i)}}{f^{(0)}}\frac{Z_0}{T_z+Z_0}\right)^2 & 0\\ 0 & \left(\frac{f^{(i)}}{f^{(0)}}\frac{Z_0}{T_z+Z_0}\right)^2 cos^2\theta\end{bmatrix}\mathbf{R_z}|_\mathbf{2}{}^{-1}(\phi). \tag{11}$$

This last equation shows that $\theta$ can be computed from the ratio of eigenvalues of $\mathbf{L_s}\mathbf{L_s}^T$, namely $(\lambda_1,\lambda_2)$,

$$cos\theta=\sqrt{\frac{\lambda_2}{\lambda_1}},$$

where

$$\lambda_1=\left(\frac{f^{(i)}}{f^{(0)}}\frac{Z_0}{Z_0+T_z}\right)^2 \tag{12}$$

is the largest eigenvalue.

The angle $\phi$ can be extracted from the eigenvectors of $\mathbf{L_s}\mathbf{L_s}^T$. The eigenvector $\mathbf{v_1}$ with largest eigenvalue equals the first column of $\mathbf{R_z}|_\mathbf{2}(\phi)$,

$$\mathbf{v_1} = \begin{bmatrix} cos\phi \\ sin\phi \end{bmatrix}.$$

At this stage, $\psi$ can be deduced by isolating $\mathbf{R_z}|_\mathbf{2}(\psi)$ in equation (10),

$$\mathbf{R_z}|_\mathbf{2}(\psi) = \frac{f^{(0)}}{f^{(i)}}(1 + \frac{T_z}{Z_0}) \begin{bmatrix} 1 & 0 \\ 0 & \frac{1}{cos\theta} \end{bmatrix} \mathbf{R_z}|_\mathbf{2}(-\phi)\mathbf{L_s}.$$

Observing, from (12), that

$$\frac{f^{(0)}}{f^{(i)}} \left(1 + \frac{T_z}{Z_0}\right) = \frac{1}{\sqrt{\lambda_1}}, \tag{13}$$

we can find $sin\psi$ and then $\psi$. Once the angles $\psi, \theta, \phi$ are known, the rotation matrix $\mathbf{R}$ can be computed as in equation (9).

From equation (13) the scaled depth is recovered as

$$1 + \frac{T_z}{Z_0} = \frac{1}{\sqrt{\lambda_1}} \frac{f^{(i)}}{f^{(0)}}.$$

This recovered depth depends on the relation between the focal lengths in consecutive frames. In robot vision applications, one may assume that the robot controls the zooming factor. Hence, the relation between focal lengths at different time instants may be assumed known even when the exact focal length is unknown.

The other two components of the translation vector can be written, from equations (4) and (12), as

$$\frac{f^{(0)}}{Z_0} \begin{bmatrix} T_x \\ T_y \end{bmatrix} = \frac{\mathbf{r} - \Delta\mathbf{u}}{\sqrt{\lambda_1}} \begin{bmatrix} \frac{1}{K_u} & 0 \\ 0 & \frac{1}{K_v} \end{bmatrix}.$$

$(K_u, K_v)$ are known from the pixel size, $\mathbf{r}$ is obtained as a tracker output and $\lambda_1$ has been deduced above as the largest eigenvalue of $\mathbf{L_s}\mathbf{L_s}^T$. Thus, we observe that the recovered scaled translation depends on the difference between the principal points in consecutive frames. This difference is usually small and depends on the changes in the focal length [4,11].

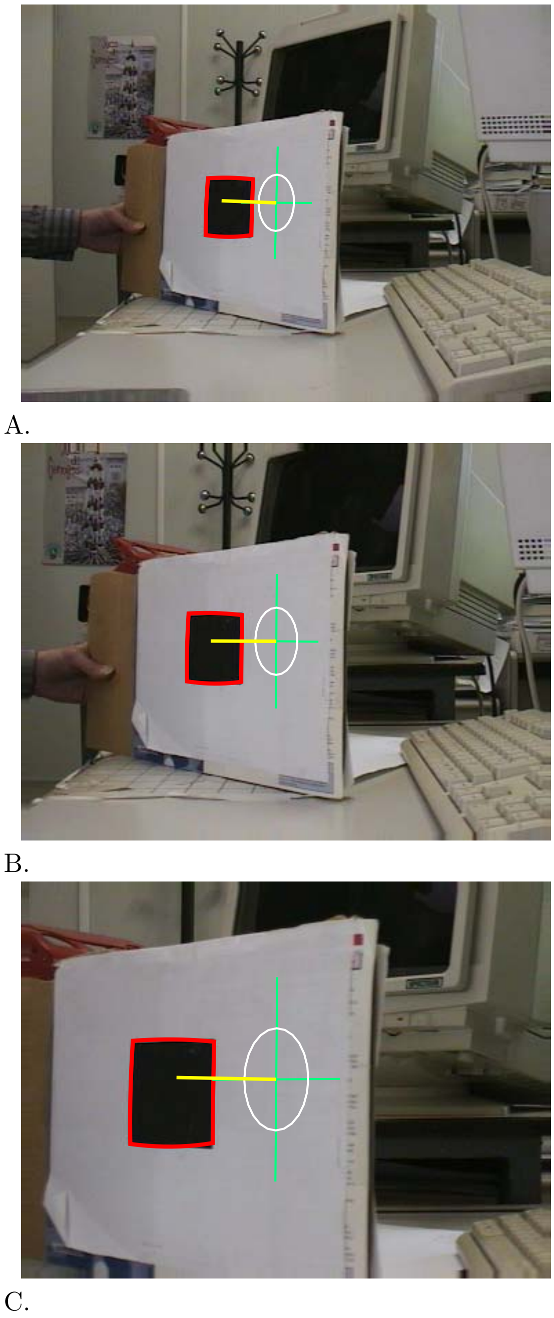
## 4. EXPERIMENTAL TESTS

Before incorporating the technique to the visual system of the robot ARGOS [25], for which it has been developed, we have performed some experiments in a more controlled setting. Two examples of the experiments conducted to test the method in the laboratory are presented. Both use an uncalibrated camera with freely varying internal parameters (i.e. focal length and principal point).

An active contour has been fitted to a printed square with sharp edges in order to ease the task of the tracker and evaluate the performance of the 3D motion estimation aside of the tracker. The estimated motion is graphically represented by a virtual object drawn in the middle of the image. The size of the virtual circle draws the estimated translation along Z, while its orientation depicts the 3D rotation.

The first experiment aims to show the zoom invariance of the recovered 3D rotation. Hence, both the target and the camera remained still while the zoom factor changed. Figure 1 shows three different frames of the sequence. As expected, the estimated 3D rotation is invariant to zooming, while the estimated translation along Z changes proportionally to the zoom factor.

Figure 2 draws a sample of the results obtained for different 3D motions and camera zooming factors. For practical convenience, the target was moved in front of the zooming camera instead of the equivalent situation, in which the target remains still while the camera moves and zooms. Again a virtual object is drawn in the middle of the screen following the motions of the target. The first image (A) is the initial view, which is taken as the template. The following frames (B to E) show the recovered motion after different movements of the target while the camera zooms in, and finally view (F) shows the estimated 3D motion when the camera zooms out. We verify that the proposed method provides qualitatively correct results.

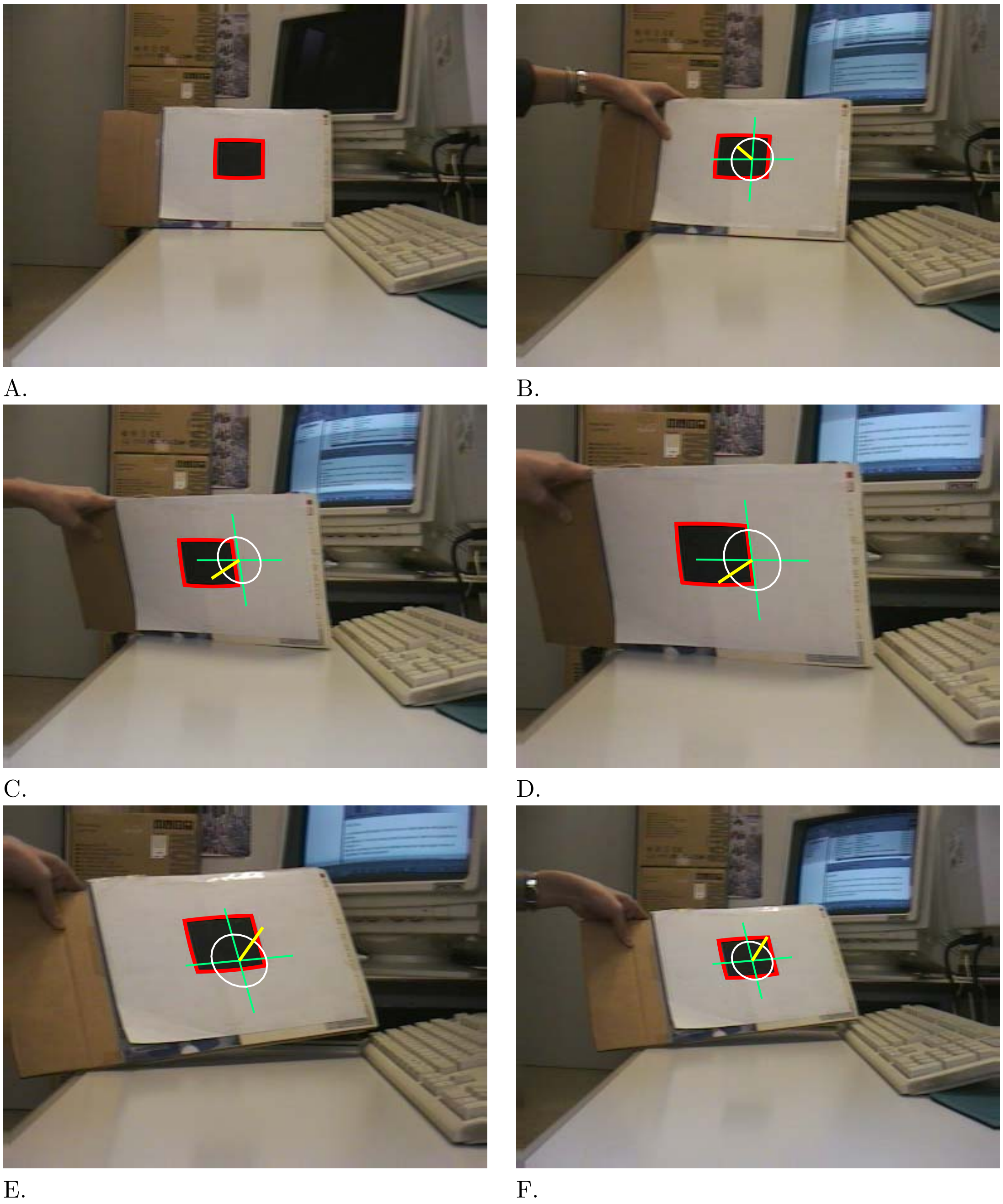


A.

B.

C.

Figure 1. *Invariance of 3D rotation recovery while zooming. The target remains still while the camera zoom factor changes. The estimated 3D rotation is invariant to zooming, while the estimated translation along Z changes proportionally to the zoom factor.*



A.

B.

C.

D.

E.

F.

Figure 2. *3D motion recovery while zooming. The first image is the initial view, which is taken as the template. The subsequent images show a virtual object drawing the recovered motion after different target movements and camera zooms.*

## 5. CONCLUDING REMARKS

We have analysed how the deformation of an active contour can be used to extract the 3D motion components while zooming. The basis of the method draws on ideas from previously published papers by the authors [21,23], and fills the remaining hole in the analysis of the deformation of an active contour for different assumptions about the intrinsic camera parameters.

The theoretical deduction along with the experimental results show that the 3D rotation matrix can be reliably recovered while zooming. However, as one could expect, the scaled depth is distorted by the camera zoom. On the other hand, the change in the principal point due to the zoom affects the recovery of the other two components of the 3D translation vector. The experiments have been conducted with a monocular camera, hence the results keep the ambiguities common in this case. However, the deduction may be easily extended to a stereo rig.

## REFERENCES

1. A. Heyden and K. Astrom. Euclidean reconstruction from image sequences with varying and unknown focal length and principal point. In *Proc. Conf. Computer Vision and Pattern Recognition*, 1997.
2. E. Hayman, T. Throhallsson, and D. Murray. Zoom-invariant tracking using points and lines in affine views an application of the affine multifocal tensors. In *Proc. 7th Int. Conf. on Computer Vision*, September 1999.
3. J.L. Crowley, P. Bobet, and C. Schmidt. Maintaining stereo calibration by tracking image points. In *Proc. Conf. Computer Vision and Pattern Recognition*, pages 483–488, 1993.
4. P. Sturm. Self calibration of a moving zoom-lens camera by pre–calibration. *Image and Vision Computing*, 15(8):583–589, 1997.
5. S. Soatto and P. Perona. Recursive estimation of camera motion from uncalibrated image sequences. In *Proc. 1st IEEE International Conference on Image Processing (ICIP)*, pages II–58–62, 1994.
6. R.Y. Tsai. A versatile camera calibration technique for high-accuracy 3d machine vision metrology using off-the-shelf tv cameras and lenses. *IEEE Journal of Robotics and Automation*, RA-3(4):323–344, 1987.
7. J. Weng, P. Cohen, and M. Herniou. Camera calibration with distortion models and accuracy evaluation. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 14(10):965–980, 1992.
8. M. Pollefeys, R. Koch, and L. Van Gool. Self-calibration and metric reconstruction in spite of varying and unknown internal camera parameters. In *Proc. 6th Int. Conf. on Computer Vision, Bombay, India*, pages 90–95, January 1998.
9. L. Agapito, R. Hartley, and E. Hayman. Linear calibration of a rotating and zooming camera. In *Proc. of the Computer Vision and Pattern Recognition Conference*, 1999.
10. R. Szeliski and P. Torr. Geometrically constrained structure from motion: Points on planes. In *Proc. of the European Workshop on 3D Structure from Multiple Images of Large Scale Environments*, pages 171–186, June 1998.
11. P. Sturm and S.J. Maybank. On plane-based camera calibration: A general algorithm, singularities, applications. In *IEEE Int. Conf. on Computer Vision and Pattern Recognition*, pages 432–437, 1999.
12. A. Bartoli, P. Sturm, and R. Horaud. Structure and motion from two uncalibrated views using points on planes. In *Third International Conference on 3D Digital Imaging and Modeling*, pages 83–90, May 2001.
13. M. Zuchelli, J. Santos-Victor, and H. I. Christensen. Constrained structure and motion estimation from optical flow. In *Proc. Int. Conf. on Pattern Recognition*, August 2002.
14. E. Malis and R. Cipolla. Camera self-calibration from unknown planar structures enforcing the multi-view constraints between collineations. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 2002.
15. B.K.P. Horn. *Robot vision.* MIT Press, 1986.
16. D.W. Murray and B.F. Buxton. *Experiments in the machine interpretation of visual mo-*

*tion*. MIT Press, 1990.

17. A. Blake, M.A. Isard, and D. Reynard. Learning to track the visual motion of contours. *J. Artificial Intelligence*, 78:101–134, 1995.
18. M.A. Isard and A. Blake. Visual tracking by stochastic propagation of conditional density. In *Proc. 4th European Conf. Computer Vision*, pages 343–356, Cambridge, England, Apr 1996.
19. A. Blake and M. Isard. *Active contours.* Springer, 1998.
20. E. Martínez and C. Torras. Integration of appearance and geometric methods for the analysis of monocular sequences. In *Proc. IST/SPIE 12th Annual Symp. on Electronic Imaging*, pages 62–70, San Jose. California. USA. January, 2000.
21. E. Martínez and C. Torras. Epipolar geometry from the deformation of an active contour. In *Proc. Int. Conference on Pattern Recognition*, pages 534–537, Barcelona, Spain. September, 2000.
22. E. Martínez and C. Torras. Qualitative vision for the guidance of legged robots in unstructured environments. *Pattern Recongnition*, 34/8:1585–1599, 2001.
23. E. Martínez and C. Torras. Depth map from the combination of matched points with active contours. In *Proc. of the IEEE International Conference on Intelligent Vehicles.*, pages 332–338, Dearborn, Michigan, USA, October, 2000.
24. E. Celaya and C. Torras. Visual navigation outdoors: the argos project. In *Proc. Int. Conference on Intelligent Autonomous Systems*, pages 63–67, Marina del Rey, California, USA, March 2002.
25. Argos Robot. *Institut de Robòtica i Informàtica (IRI).* www-iri.upc.es/people/porta/robots/argos/index.html.