Epipolar Geometry from the Deformation of an Active Contour

Elisa Martínez
Dep. de Comunicacions i Teoria del Senyal
Enginyeria La Salle. Universitat Ramon Llull
Pge. Bonanova, 8, 08022 Barcelona, Spain.
elisa@salleurl.edu http://www.salleurl.edu/˜elisa

Carme Torras
Institut de Robòtica i Informàtica Industrial.
Gran Capità, 2-4, 08034 Barcelona, Spain.
ctorras@iri.upc.es http://www-iri.upc.es/people/torras

Abstract

An active contour is used to track a target in a sequence recorded by a walking robot in an unstructured scene . The deformations of the contour are analysed in order to extract the robot's egomotion, from which we compute the epipolar geometry that guides the matching between different views of the scene. The results prove that the proposed solution is a promising alternative to the prevalent techniques based on the costly computation of displacement or velocity fields [12, 14].

1. Introduction

This paper is part of a project aimed at guiding a walking robot towards a visual target in an unstructured outdoor scene. The robot has six legs and it is equipped with a compass and a single camera rigidly mounted on its body. We assume that the internal parameters of the camera can be calibrated off-line. Once an operator marks a static target on an image recorded by the camera, the robot has to reach the target as autonomously as possible.

An active contour is used to track the target in the sequence recorded by the walking robot. Active contours have proved to be an excellent solution for real time tracking [1]. They have also been used to estimate surface orientation and time to contact with a target when the viewer can make deliberate movements or has stereoscopic vision [4]. Here we propose to analyse the deformations of the active contour in order to extract the viewer egomotion (3D rotation and scaled translation), from which we compute the epipolar geometry that relates different frames of the sequence. The latter allows to guide the search for matches between frames, and the structure of the scene can be computed from point matches as usual in stereo systems [5]. The structure helps the robot to avoid obstacles and keep a feature map of the environment.

The deduction is here tailored to the particular features of our application. A general analysis of the relation between the deformations of an active contour fitted to a target and the egomotion of a freely moving camera that observes it was presented in a previous work [9], along with a discussion of the limitations and ambiguities of the method. In the present work, due to the balances of the legged robot [3], the optical axis is assumed to be normal to the gravity vector. Which, together with the information provided by the compass, allows to avoid the rotation–translation ambiguity common in weak–perspective camera models when the axis of rotation is located in the image plane. Avoiding this ambiguity, the epipolar geometry can be recovered in real time under the same framework used to track the target.

The analysis of image sequences to estimate camera motion and epipolar geometry became a key theme in computer vision research during the past decade [10, 12, 14, 2]. The usual approach is based on optic flow, either by obtaining the velocity vectors at all image positions, or by extracting some salient points and computing their displacement vectors from frame to frame. Both procedures are computationally costly. Moreover, the epipolar lines relating two different views of a scene are usually computed from the essential matrix, which is extracted from a set of point matches. Thus, the extraction of the epipolar lines needed to guide the matches between frames is generally based on initial matches. The proposed solution breaks this loop and offers a computationally effective alternative.

2 Projection of 3D motion onto the image plane

A static object in 3D space is used as reference to estimate the camera motion. We fit a closed curve to its occluding contour in the initial position, which 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 curve is traversed. The projection of $\mathbf{D_0}(s)$ on the image plane is called the template, $\mathbf{d_0}(s)$. 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 a weak perspective situation, i.e. when the object fits in a small field of view and has a small range of depths compared to its distance to the camera, then 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 assumptions 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. We calculate the projected curve using a weak–perspective camera model.

Due to the balances of the legged robot [3], the optical axis is kept normal to the gravity vector and the rotation of the camera is reduced to a rotation around the Y axis. Then,

$$\mathbf{R} = \begin{bmatrix} cos\psi & 0 & -sin\psi \\ 0 & 1 & 0 \\ sin\psi & 0 & cos\psi \end{bmatrix}.$$

Taking the camera coordinate frame as reference, $Z_0(s)$ can be approximated by the average depth Z_0 of the contour, hence the projected curve on the image plane has the following expression,

$$\mathbf{d}(s) = \frac{f}{\mathbf{R_3}\mathbf{D_0}(s) + T_z}\left(\begin{bmatrix}\mathbf{R_1}\\ \mathbf{R_2}\end{bmatrix}\begin{bmatrix}X_0(s)\\ Y_0(s)\\ Z_0\end{bmatrix} + \begin{bmatrix}T_x\\ T_y\end{bmatrix}\right), \tag{2}$$

where f is the focal length, $\mathbf{R_i}$ is the i–th row of the rotation matrix $\mathbf{R}$, and $\mathbf{T} = (T_x, T_y, T_z)^T$.

Without loss of generality, we can assume that the centroid of the template $\mathbf{d_0}(s)$ equals the principal point. Thus, under weak perspective, $R_{31}X_0(s) + R_{32}Y_0(s) \ll R_{33}Z_0 + T_z$, and equation (2) can be rewritten as

$$\mathbf{d}(s) = \frac{f}{cos\psi Z_0 + T_z}\left(\begin{bmatrix} cos\psi & 0\\ 0 & 1\end{bmatrix}\begin{bmatrix}X_0(s)\\ Y_0(s)\end{bmatrix} + \right.$$
$$\left. + Z_0\begin{bmatrix}-sin\psi\\ 0\end{bmatrix} + \begin{bmatrix}T_x\\ T_y\end{bmatrix}\right). \tag{3}$$

In particular, the template is

$$\mathbf{d_0}(s) = \frac{f}{Z_0}\begin{bmatrix}X_0(s)\\ Y_0(s)\end{bmatrix}. \tag{4}$$

Combining equations (3) and (4), the difference between the curve at a particular instant and the template is

$$\mathbf{d}(s) - \mathbf{d_0}(s) = (\mathbf{M} - \mathbf{I})\mathbf{d_0}(s) + \mathbf{t}, \tag{5}$$

where $\mathbf{I}$ is the identity matrix,

$$\mathbf{M} = \frac{Z_0}{cos\psi Z_0 + T_z}\begin{bmatrix} cos\psi & 0\\ 0 & 1\end{bmatrix}, \tag{6}$$

and

$$\mathbf{t} = \frac{f}{cos\psi Z_0 + T_z}\left(Z_0\begin{bmatrix}-sin\psi\\ 0\end{bmatrix} + \begin{bmatrix}T_x\\ T_y\end{bmatrix}\right). \tag{7}$$

This result shows that the rigid motion of the 3D curve (equation (1)) projects as an affine deformation of the template onto the image plane (equation (5)), when the curve is viewed under weak perspective.

3 Template deformation from the analysis of active contours

In this section we review how the deformation of the template in the image plane can be recovered from the analysis of an active contour fitted to it. The contour is represented as a parametric spline curve $\mathbf{d}(s) = (d_x(s), d_y(s))^T$, where both $d_x(s)$ and $d_y(s)$ are B-spline curves. We can write them as a function of their control points,

$$\mathbf{d}(s) = \begin{bmatrix}\mathbf{B}(s)\mathbf{Q^x}\\ \mathbf{B}(s)\mathbf{Q^y}\end{bmatrix} = \begin{bmatrix}\mathbf{B}(s) & \mathbf{0}^T\\ \mathbf{0}^T & \mathbf{B}(s)\end{bmatrix}\begin{bmatrix}\mathbf{Q^x}\\ \mathbf{Q^y}\end{bmatrix} = \mathbf{U}(s)\mathbf{Q}, \tag{8}$$

where $\mathbf{Q^i}$ is a column vector with the i–th components of the control points, $\mathbf{B}(s)$ is a row vector of B-spline basis functions [7, 1], $\mathbf{U}(s) = \mathbf{I} \otimes \mathbf{B}(s)$[1], $\mathbf{Q}$ is the vector of control points and $\mathbf{0}$ is a column vector of zeros. In particular, the template can be written as

$$\mathbf{d_0}(s) = \mathbf{U}(s)\mathbf{Q_0}, \tag{9}$$

where $\mathbf{Q_0}$ is the vector of control points for the template. Substituting equations (8) and (9) in equation (5), and using the convex hull property of B–spline curves ($\mathbf{B}(s)\mathbf{1} = 1$), we obtain that the difference between the control points is,

$$\mathbf{Q} - \mathbf{Q_0} = \mathbf{W}\mathbf{X},$$

[1] $\otimes$ is the kronecker product.

where **W** is the shape matrix,

$$\mathbf{W} = \left(\begin{bmatrix} \mathbf{1} \\ \mathbf{0} \end{bmatrix}, \begin{bmatrix} \mathbf{0} \\ \mathbf{1} \end{bmatrix}, \begin{bmatrix} \mathbf{Q_0^x} \\ \mathbf{0} \end{bmatrix}, \begin{bmatrix} \mathbf{0} \\ \mathbf{Q_0^y} \end{bmatrix} \right)$$

and **X** is the *shape vector*

$$\mathbf{X} = (t_x, t_y, M_{11} - 1, M_{22} - 1)^T,$$

t_i are the components of $\mathbf{t}$ and M_{ij} are the elements of **M**.

We use the active contour tracker of Blake *et al.* [1], based on the Kalman filter, to compute the shape vector **X** along the sequence. The active contour is forced to lie in the space of affine deformations of the template for each frame.

4 Egomotion computation

Our purpose now is to compute the 3D motion parameters from the deformation of the curve in the image plane. From the shape vector, using equations (6) and (7) we obtain,

$$cos\psi = \frac{M_{11}}{M_{22}} \tag{10}$$

$$\frac{T_x}{Z_0} = \frac{t_x}{fM_{22}} + sin\psi \tag{11}$$

$$\frac{T_y}{Z_0} = \frac{t_y}{fM_{22}} \tag{12}$$

$$\frac{T_z}{Z_0} = \frac{1}{M_{22}} - cos\psi \tag{13}$$

These results keep the ambiguities usual in monocular images. Equations (11), (12), (13) show the effect of the scale–depth ambiguity in the computation of the translation. There is no way to recover the absolute translation. Equation (10) keeps the Necker reversal ambiguity. From $cos\psi$ only the magnitude of ψ can be computed. The sign of the angle cannot be recovered. However, this ambiguity can be solved assuming continuity in the motion.

Another ambiguity appears, namely the rotation–translation ambiguity, which is common when the axis of rotation is located in the image plane. The ambiguity arises because rotation about the Y axis and translation along the X axis produce similar effects as reflected in equation (11) [6, 12]. This ambiguity is responsible for the invariance of $\frac{M_{11}}{M_{22}}$ to small changes in ψ. However, this is not a problem in our application, since the robot is equipped with a compass that provides the ψ angle. The bas-relief ambiguity is also cancelled when the rotation is known.

5 Computation of epipolar geometry

In the preceding sections, we have been working with a simplified camera model as we were focusing the processing on the target. Now, we switch to a more general camera model to compute the epipolar geometry of the whole image. The weak–perspective camera is adequate to model the imaging process of the target, but it does not generally fit the rest of the image, particularly when the scene has objects at different depths.

A point $\mathbf{u}^{(1)}$ in the first image corresponds to a 3D point that lies on the ray that backprojects through $\mathbf{u}^{(1)}$. Its corresponding point in the second image, $\mathbf{u}^{(2)}$, lies on the projection of this ray, namely, the epipolar line of $\mathbf{u}^{(1)}$. The epipolar lines are usually computed from the essential matrix **E** [8, 13]. It relates the projections $\mathbf{u}^{(1)}, \mathbf{u}^{(2)}$ of a 3D point, in homogeneous notation, as follows,

$$\mathbf{u}^{(2)T}\mathbf{E}\mathbf{u}^{(1)} = 0. \tag{14}$$

E can be split up [8, 13, 11] as

$$\mathbf{E} = [\mathbf{T}]_*\mathbf{R}$$

where **R** is the rotation matrix and $[\mathbf{T}]_*$ is a matrix obtained from the elements of $\mathbf{T}$,

$$[\mathbf{T}]_* = \begin{bmatrix} 0 & -T_z & T_y \\ T_z & 0 & -T_x \\ -T_y & T_x & 0 \end{bmatrix}$$

E is generally computed from point matches; here we propose an alternative. **E** is defined only up to a scale factor, as observed in equation (14), therefore it can be computed from the scaled translation vector $\frac{\mathbf{T}}{Z_0}$ obtained in the preceding section.

Using homogeneous notation, a line $\mathbf{l}^{(2)}$ passing through a point $\mathbf{u}^{(2)}$ fulfils the following equation [5],

$$\mathbf{u}^{(2)}{}^{\mathbf{T}}\mathbf{l}^{(2)} = 0.$$

Therefore, from equation (14), the epipolar line can be computed as

$$\mathbf{l}^{(2)} = \mathbf{E}\mathbf{u}^{(1)}$$

The epipolar lines have been computed to be used as a guide for matching features between frames. Some results are shown in Figs. 1 and 2. Fig. 1 shows an image of the sequence, in which an active contour has been fitted to the target, namely the door. A simple target was chosen, although the capability of active contours to track complex shapes and their robustness to occlusions have been proved elsewhere [1]. Some salient features have been numbered, and their epipolar lines have been computed to guide their

Figure 1. Initial image.

Figure 2. Epipolar lines.

matching with salient features in Fig. 2. The epipolar lines are drawn in Fig. 2. The local processing makes the results invariant to independent motions in the scene. Moreover, the robustness of active contours to partial occlusions of the target is transfered to the proposed method.

6 Concluding remarks

This paper has presented a novel approach for the computation of the epipolar geometry from the deformations of an active contour fitted to a target, which is observed by a walking robot. The prevailing trend relies on point matches between frames, which is limited to non homogeneous regions where some salient points can be detected. Experimental results confirm that the analysis of the contour is a good solution on its own to estimate egomotion; and it can be taken, in general, as an interesting complement to the analysis of point matches in the scene, since contours can be easily fitted to homogeneous regions and provide the epipolar constraints needed to guide point matching between frames. The recovery of 3D structure is accelerated as the epipolar constraints are provided in real time by the active contour (25 frames per second with a Silicon Graphics Indy at 150 MHz). Note that the computation time is independent of the image size because of the local processing.

The proposed solution is limited to those scenes in which a target is visible under weak perspective; however, it is a common situation when trying to guide a robot towards a distant target. The method can be directly extended to a stereo vision system, which will avoid the ambiguities common in monocular sequences. Future work is planed to test the performance of the method using a stereo rig.

Acknowledgments. This research has been partially supported by the grant CICYT TAP97-1209 of the Spanish Science and Technology Council.

References

[1] A. Blake and M. Isard. *Active contours*. Springer, 1998.

[2] A. Branca, E. Stella, and A. Distante. Passive navigation using egomotion estimates. *Image and Vision Computing*, 18:833–841, 2000.

[3] E. Celaya and F. Porta. A control structure for the locomotion of a legged robot on difficult terrain. *IEEE Robotics and Automation Magazine*, 5:43–51, 1998.

[4] R. Cipolla and A. Blake. Surface orientation and time to contact from image divergence and deformation. In *Proc. 2nd European Conf. Computer Vision*, pages 187–202, 1992.

[5] O. Faugeras. *3D Computer Vision*. MIT Press, 1993.

[6] C. Fermuller and Y. Aloimonos. Qualitative egomotion. *Int. J. Computer Vision*, 15:7–29, 1995.

[7] J. Foley, A. van Dam, S. Feiner, and F. Hughes. *Computer Graphics. Principles and practice*. Addison-Wesley Publishing Company, 1996.

[8] H. Longuet-Higgins. A computer algorithm for reconstructing a scene from two projections. *Nature*, 293:133–135, 1981.

[9] 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, January 2000.

[10] P. McLauchlan and D. Murray. A unifying framework for structure and motion recovery from image sequences. In *Proc. 5th Int. Conf. on Computer Vision, Cambridge, MA*, pages 314–320, 1995.

[11] J. Mundy and A. Zisserman. Projective geometry for machine vision. In J. Mundy and A. Zisserman, editors, *Geometric Invariance in Computer Vision*. MIT Press, 1992.

[12] T. Tian, C. Tomasi, and D. Heeger. Comparison of approaches to egomotion computation. In *Proc. Conf. Computer Vision and Pattern Recognition*, pages 315–320, 1996.

[13] R. Tsai and T. Huang. Uniqueness and estimation of three-dimensional motion parameters of rigid objects with curved surfaces. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 6:13–27, 1984.

[14] Z. Zhang. Determining the epipolar geometry and its uncertainty: A review. Technical report, INRIA, Sophia Antipolis, France, 1996. No. 2927.