PERGAMON

Pattern Recognition 34 (2001) 1585–1599

PATTERN RECOGNITION
THE JOURNAL OF THE PATTERN RECOGNITION SOCIETY

www.elsevier.com/locate/patcog

Qualitative vision for the guidance of legged robots in unstructured environments

Elisa Martínez[a,*], Carme Torras[b]

[a]Departament de Comunicacions i Teoria del Senyal, Enginyeria La Salle, Pge. Bonanova, 8, 08022 Barcelona, Spain
[b]Institut de Robòtica i Informàtica Industrial (CSIC-UPC). Gran Capità, 2-4, 08034 Barcelona, Spain

Received 15 May 2000; accepted 15 May 2000

Abstract

Visual procedures especially tailored to the constraints and requirements of a legged robot are presented. They work for an uncalibrated camera, with pan and zoom, freely moving towards a stationary target in an unstructured environment that may contain independently moving objects. The goal is to dynamically analyse the image sequence in order to extract information about the robot motion, the target position and the environment structure. The development is based on the deformations of an active contour fitted to the target. Experimental results confirm that the proposed approach constitutes a promising alternative to the prevailing trend based on the costly computation of displacement or velocity fields. © 2001 Pattern Recognition Society. Published by Elsevier Science Ltd. All rights reserved.

Keywords: Image sequences; Egomotion recovery; 3D reconstruction; Uncalibrated camera; Unstructured environment; Contour tracking; Visual robot navigation

1. Introduction

The present work is part of a project aimed at the development of a low-cost walking robot for exploratory tasks [1,2]. It is a six-legged robot with three degrees of freedom (dof) per leg, and it is equipped with a compass and a single camera with one dof (pan). Once an operator marks a given target on an image captured by the camera, the robot has to reach the target as autonomously as possible. Since our robot has deliberately limited resources, we do not look for very sophisticated procedures aimed at attaining 100% performance success, but instead we like to reach the best possible compromise between simplicity and performance. Then the accuracy demands are low for this application but, as a counterpart, many constraints are imposed on the process of estimating camera motion:

- monocular vision,
- uncalibrated camera, with pan and zoom,
- unstructured environment,
- unknown camera motion,
- visual control through robot legs and pan of the camera,
- limited computational resources,
- medium time demands.

The calibration parameters of a camera mounted on a mobile robot are likely to change over time. Although some intrinsic camera parameters (e.g., pixel size and aspect ratio) remain constant for long periods of time [3], others (e.g., image centre and focal length) may change drastically along an image sequence [4]. The process of calibration with the aid of a calibration pattern is inapplicable in cases where the camera optical parameters undergo frequent changes. Different approaches have recently emerged that consist in autocalibrating the camera on-line [5] or in designing methods which do not need the calibration parameters [6]. Among the latter,

* Corresponding author. Tel.: +34-93-290-2440; fax: +34-93-290-2416.
E-mail addresses: elisa@salleurl.edu (E. Martínez), ctorras@iri.upc.es (C. Torras).

PII: S0031-3203(00)00093-5

the procedures developed in this paper highlight the abilities of a vision system based on an uncalibrated camera. The same algorithms would supply qualitatively different information depending on the degree of camera calibration [7].

There are many works dealing with the visual guidance of robots in structured scenes [8,9]; less works address the visual guidance of wheeled or tracked robots in slightly structured or nonstructured environments [10–12]; but works tackling the visual guidance of walking robots in unstructured scenes are very scarce [13,14]. In this paper, we describe procedures that provide a qualitative estimation of robot motion, target position and environment structure. Other visual processes related to landmark detection and recognition are described elsewhere, as are the aspects related to locomotion and navigation within our project.

Estimating camera motion and scene structure from a sequence of images has been the object of intense research within the computer vision community for some years now [15–21]. The usual approach is based on optic flow. This can be computed in two ways, either by obtaining the velocity vectors at all image positions, or by extracting some clearly distinguishable features and tracking them from frame to frame. Both procedures are computationally costly and they fail to provide results in homogeneous regions.

There are a few works that compute egomotion on the basis of only local information. Cipolla and Blake [22] use the area moments of closed contours to estimate surface orientation and time-to-contact with a target. Their procedure can be used for qualitative visual navigation, if the viewer can make deliberate movements or has stereoscopic vision. For a legged robot it is practically impossible to change the position and orientation of its body in a predefined way, as arm robots do with their end-effectors. What a legged robot can do is to always try to maintain its body in a reference position (say, horizontally) irrespective of terrain orientation, by means of the so-called "balances" [2], a thing that tracked robots cannot do without extra degrees of freedom in the camera subsystem.

The method here proposed exploits the particular features of our application to simplify the estimation process, so that it can be performed under the constraints listed above. The proposed scheme is summarized in Fig. 1. The method combines the analysis of active contours [23] with the geometric constraints between different views of a single scene, namely the epipolar geometry. An active contour is automatically fitted to a static target marked by the operator in the image, from which a shape vector is extracted for each frame. The analysis of the active contour provides a direct measure of image deformation that allows one to compute the egomotion up to a scale factor and the time-to-contact, which is a qualitative measure of the distance to the target. To extract

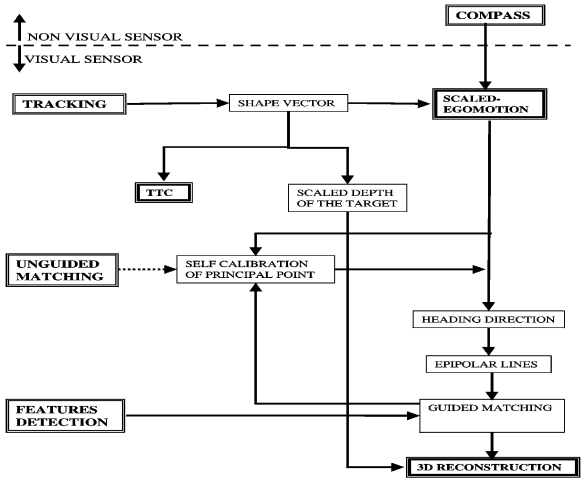

Fig. 1. Global scheme.

information about the scene structure, the analysis of the active contour is combined with the data provided by point matches between the different frames. We prove that this combination makes it possible to self-calibrate the principal point, which is used to compute the heading direction, or epipole, from the scaled egomotion. Once the principal point is known, the epipolar geometry can be directly extracted from the active contour. The epipolar constraints guide the matching between salient points in two different views of the scene. Finally, combining the matched points with the scaled egomotion, the qualitative 3D scene structure is recovered by interpolating the depth of the matched points and the depth of the points inside the target.

The paper is structured as follows. The next section characterizes the projection of a moving curve under an affine camera model. Section 3 presents the derivation of the shape vector, from which we extract both the egomotion (Section 4) and the time to contact with the target (Section 5). The subsequent computation of the heading direction or epipole is described in Section 6, where we also explain how the principal point can be self-calibrated by combining a set of point matches with the analysis of the shape vector. Section 7 is devoted to the recovery of the epipolar geometry, which is used to match different views. The combination of the analysis of the contour with the matches allows one to recover a qualitative depth map (Section 8). Finally, the advantages and limitations of the proposed procedure are discussed in Section 9.

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 $\mathbf{D}_0(s) = (X_0(s), Y_0(s), Z_0(s))^{\mathrm{T}}$ where s is a parameter that increases as the curve is traversed. The projection of $\mathbf{D}_0(s)$ onto 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 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 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 an affine camera model. The affine camera, introduced by Mundy and Zisserman [24], is a generalization of orthographic, weak perspective and paraperspective projections. This is an approximation to the full perspective, equivalent to a weak perspective camera with unknown internal calibration parameters.

Taking the camera coordinate frame as reference, $Z_0(s)$ can be approximated by the average depth Z_0 of the contour. This approximation can be made because we have already assumed a weak perspective situation in which the target is shallow with respect to its distance to the camera. We also assume that the occluding contour of the target is contained in a plane parallel to the image plane, otherwise the results would suffer from the bas-relief ambiguity. Then the projected curve on the image plane has the following expression:

$$\mathbf{d}(s) = \begin{bmatrix} K_u & 0 \\ 0 & -K_v \end{bmatrix} \frac{f}{\mathbf{R}_3\mathbf{D}_0(s)} \times \left(\begin{bmatrix} R_{11} & R_{12} & R_{13} \\ R_{21} & R_{22} & R_{23} \end{bmatrix} \begin{bmatrix} X_0(s) \\ Y_0(s) \\ Z_0 \end{bmatrix} + \begin{bmatrix} T_x \\ T_y \end{bmatrix} \right) + \begin{bmatrix} u_0 \\ v_0 \end{bmatrix}, \tag{2}$$

where f is the focal length, $K_u \times K_v$ is the pixel size, (u_0, v_0) is the principal point, R_{ij} are the elements of the rotation matrix $\mathbf{R}$, $\mathbf{R}_3$ is the third row of $\mathbf{R}$, and $\mathbf{T} = (T_x, T_y, T_z)^{\mathrm{T}}$. We assume that the calibration parameters f, K_u, K_v, u_0, v_0 are unknown, as corresponds to an affine camera model. However, we explicitly write the calibration parameters in order to highlight their effect on different measures. We will finally prove that we can provide the robot with enough information for navigation without a priori knowledge of the calibration parameters.

Without loss of generality, we can assume that the centre of $\mathbf{D}_0(s)$ has $X = Y = 0$ components; it is equivalent to assuming that the centre 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 Eq. (2) can be rewritten as

$$\mathbf{d}(s) = \begin{bmatrix} K_u & 0 \\ 0 & -K_v \end{bmatrix} \frac{f}{R_{33}Z_0 + T_z} \times \left(\begin{bmatrix} R_{11} & R_{12} \\ R_{21} & R_{22} \end{bmatrix} \begin{bmatrix} X_0(s) \\ Y_0(s) \end{bmatrix} + Z_0 \begin{bmatrix} R_{13} \\ R_{23} \end{bmatrix} \begin{bmatrix} T_x \\ T_y \end{bmatrix} \right) + \begin{bmatrix} u_0 \\ v_0 \end{bmatrix}. \tag{3}$$

In particular, the projection of the template is

$$\mathbf{d}_0(s) = \frac{f}{Z_0} \begin{bmatrix} K_u & 0 \\ 0 & K_v \end{bmatrix} \begin{bmatrix} X_0(s) \\ Y_0(s) \end{bmatrix} + \begin{bmatrix} u_0 \\ v_0 \end{bmatrix}. \tag{4}$$

Combining Eqs. (3) and (4),

$$\mathbf{d}(s) - \begin{bmatrix} u_0 \\ v_0 \end{bmatrix} = \frac{Z_0}{R_{33}Z_0 + T_z} \begin{bmatrix} K_u & 0 \\ 0 & -K_v \end{bmatrix} \begin{bmatrix} R_{11} & R_{12} \\ R_{21} & R_{22} \end{bmatrix} \times \begin{bmatrix} \frac{1}{K_u} & 0 \\ 0 & \frac{-1}{K_v} \end{bmatrix} \left(\mathbf{d}_0(s) - \begin{bmatrix} u_0 \\ v_0 \end{bmatrix} \right) + \frac{f}{R_{33}Z_0 + T_z} \times \begin{bmatrix} K_u & 0 \\ 0 & -K_v \end{bmatrix} \left(Z_0 \begin{bmatrix} R_{13} \\ R_{23} \end{bmatrix} + \begin{bmatrix} T_x \\ T_y \end{bmatrix} \right).$$

Now, it is interesting to observe that

$$\left(\mathbf{d}_0(s) - \begin{bmatrix} u_0 \\ v_0 \end{bmatrix} \right)$$

is the template centred on the upper left corner of the image. Thus it can be computed from the observed template by subtracting the coordinates of its centre.

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

$$\mathbf{d}(s) - \mathbf{d}_0(s) = (\mathbf{L} - \mathbf{I}) \left(\mathbf{d}_0(s) - \begin{bmatrix} u_0 \\ v_0 \end{bmatrix} \right) + \mathbf{p}, \tag{5}$$

where $\mathbf{I}$ is the 2×2 identity matrix,

$$\mathbf{L} = \frac{Z_0}{R_{33}Z_0 + T_z} \begin{bmatrix} R_{11} & -R_{12}\frac{K_u}{K_v} \\ -R_{21}\frac{K_v}{K_u} & R_{22} \end{bmatrix}, \tag{6}$$

$$\mathbf{p} = \frac{1}{R_{33}Z_0 + T_z}\begin{bmatrix}\alpha_u & 0\\ 0 & \alpha_v\end{bmatrix}\left(Z_0\begin{bmatrix}R_{13}\\ R_{23}\end{bmatrix} + \begin{bmatrix}T_x\\ T_y\end{bmatrix}\right) \tag{7}$$

and $\alpha_u = fK_u, \alpha_v = -fK_v$.

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

3. Affine deformation from the analysis of active contours

In this section we explain how the affine deformation of the template in the image plane can be recovered from the analysis of an active contour fitted to it.

A contour can be represented as a parametric spline curve, as common in Computer Graphics [25], $\mathbf{d}(s) = (d_x(s), d_y(s))^{\mathrm{T}}$, where both $d_x(s)$ and $d_y(s)$ are B-spline curves. We can write them as a function of their control points,

$$d_x(s) = \mathbf{B}(s)\mathbf{Q}^x, \quad d_y(s) = \mathbf{B}(s)\mathbf{Q}^y,$$

where $\mathbf{Q}^i$ is a column vector of control points for the ith component and $\mathbf{B}(s)$ is a row vector of B-spline basis functions [25,23].

Putting both expressions together, we obtain a compact expression for $\mathbf{d}(s)$

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

where $\mathbf{U}(s) = \mathbf{I}\otimes\mathbf{B}(s)$[1] and $\mathbf{Q}$ is the vector of control points. In particular, the template can be written as

$$\mathbf{d}_0(s) = \mathbf{U}(s)\mathbf{Q}_0.$$

Substituting this expression into Eq. (5), we obtain

$$\mathbf{d}(s) - \mathbf{d}_0(s) = (\mathbf{L} - \mathbf{I})\mathbf{U}(s)\mathbf{Q}_0 + \mathbf{p},$$

where $\mathbf{Q}_0$ is the vector of control points of the observed template minus the coordinates of its centre. Observing that $\mathbf{B}(s)\mathbf{1} = 1$ from the convex hull property of B-spline curves, and using Eq. (8), the difference between $\mathbf{d}(s)$ and $\mathbf{d}_0(s)$ can be rewritten as

$$\begin{aligned}\mathbf{d}(s) - \mathbf{d}_0(s) = {} & p_x\mathbf{U}(s)\begin{bmatrix}\mathbf{1}\\ \mathbf{0}\end{bmatrix} + p_y\mathbf{U}(s)\begin{bmatrix}\mathbf{0}\\ \mathbf{1}\end{bmatrix}\\ & + (L_{11} - 1)\mathbf{U}(s)\begin{bmatrix}\mathbf{Q}_0^x\\ \mathbf{0}\end{bmatrix} + L_{12}\mathbf{U}(s)\begin{bmatrix}\mathbf{Q}_0^y\\ \mathbf{0}\end{bmatrix}\\ & + L_{21}\mathbf{U}(s)\begin{bmatrix}\mathbf{0}\\ \mathbf{Q}_0^x\end{bmatrix} + (L_{22} - 1)\mathbf{U}(s)\begin{bmatrix}\mathbf{0}\\ \mathbf{Q}_0^y\end{bmatrix}.\end{aligned}$$

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

Comparing this result with expression (8), we can conclude that the difference in control points $\mathbf{Q} - \mathbf{Q}_0$ can be written as a linear combination of six vectors. Therefore, using matrix notation,

$$\mathbf{Q} - \mathbf{Q}_0 = \mathbf{W}\mathbf{X},$$

where $\mathbf{W}$ is the shape matrix with the six vectors as columns,

$$\mathbf{W} = \left(\begin{bmatrix}\mathbf{1}\\ \mathbf{0}\end{bmatrix}, \begin{bmatrix}\mathbf{0}\\ \mathbf{1}\end{bmatrix}, \begin{bmatrix}\mathbf{Q}_0^x\\ \mathbf{0}\end{bmatrix}, \begin{bmatrix}\mathbf{0}\\ \mathbf{Q}_0^y\end{bmatrix}, \begin{bmatrix}\mathbf{0}\\ \mathbf{Q}_0^x\end{bmatrix}, \begin{bmatrix}\mathbf{Q}_0^y\\ \mathbf{0}\end{bmatrix}\right) \tag{9}$$

and $\mathbf{X}$ is a vector with the six parameters of the linear combination, namely the *shape vector*,

$$\mathbf{X} = (p_x, p_y, L_{11} - 1, L_{22} - 1, L_{21}, L_{12})^{\mathrm{T}}.$$

We use the active contour tracker of Blake et al. [26], which is based on the Kalman filter, to compute the shape vector $\mathbf{X}$ along the sequence. The active contour is forced to lie in the space of affine deformations of the template for each frame.

4. 3D egomotion recovery

As mentioned in the introduction, due to the balances of the legged robot [2], 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,

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

and

$$\mathbf{L} = \frac{Z_0}{Z_0\cos\psi + T_z}\begin{bmatrix}\cos\psi & 0\\ 0 & 1\end{bmatrix}, \tag{10}$$

$$\mathbf{p} = \frac{1}{Z_0\cos\psi + T_z}\begin{bmatrix}\alpha_u & 0\\ 0 & \alpha_v\end{bmatrix}\begin{bmatrix}T_x - Z_0\sin\psi\\ T_y\end{bmatrix} \tag{11}$$

and the shape vector recovered from the tracking of the contour is

$$\begin{aligned}\mathbf{X} = \Bigg(& \frac{\alpha_u(T_x - Z_0\sin\psi)}{Z_0\cos\psi + T_z}, \frac{\alpha_v T_y}{Z_0\cos\psi + T_z},\\ & \frac{Z_0}{Z_0\cos\psi + T_z}\cos\psi - 1, \frac{Z_0}{Z_0\cos\psi + T_z} - 1, 0, 0\Bigg).\end{aligned}$$

Our purpose now is to compute the 3D motion parameters from the affine deformation of the curve in the image plane. From the shape vector we directly obtain

the rotation angle ψ up to a reversal,

$$\cos\psi = \frac{L_{11}}{L_{22}} \tag{12}$$

and the scaled components of the translation vector

$$\alpha_u \frac{T_x}{Z_0} = \frac{p_x}{L_{22}} + \alpha_u \sin\psi, \tag{13}$$

$$\alpha_v \frac{T_y}{Z_0} = \frac{p_y}{L_{22}}, \tag{14}$$

$$\frac{T_z}{Z_0} = \frac{1}{L_{22}} - \cos\psi. \tag{15}$$

These results keep the ambiguities usual in monocular images. Eqs. (13)–(15) show the effect of the scale-depth ambiguity in the computation of the translation. There is no way to recover the absolute translation; only the scaled translation can be computed. Eq. (12) keeps the Necker reversal ambiguity. From $\cos\psi$ only the magnitude of ψ can be computed. The sign of the angle cannot be recovered.

The bas-relief ambiguity is only cancelled if we assume that the occluding contour of the object is in a plane parallel to the image plane in the initial frame. Therefore, the angles are measured taking into account this assumption. However, 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 Eq. (13). The translation along the X-axis is added to $\alpha_u \sin\psi$, and the two terms cannot be split unless one of them is known.

This ambiguity is responsible for the invariance of L_{11}/L_{22} to changes in ψ. As far as the change in ψ does not cause a sufficient change in perspective, the projected curve is nearly the same as the one we would have observed if the camera had translated along X. Fermuller and Aloimonos [27] explain this ambiguity by proving that the images of points rotating around the Y-axis of the camera describe hyperbolas whose major axes coincide with the X-axis of the image plane. Therefore, the ambiguity arises specially when a weak perspective or affine camera model is used. It can only be avoided if the whole image does not fit in the weak perspective model and a nonlocal processing is applied (as proposed in Ref. [28]), or there are motion parallax effects in the observed regions [29]. A comparison between these methods is presented in Ref. [19].

Since the method proposed in this paper is based on a local processing, it is unable to solve the rotation–translation ambiguity. However, this is not a problem in our application, since the robot is equipped with a compass that provides the ψ angle and the camera is able to pan. This way, the camera can compensate the rotation detected by the compass and provide an image free of rotation. In this case, the shape vector becomes

$$\mathbf{X} = \left(\frac{\alpha_u T_x}{Z_0 + T_z}, \frac{\alpha_v T_y}{Z_0 + T_z}, \frac{Z_0}{Z_0 + T_z} - 1, \frac{Z_0}{Z_0 + T_z} - 1, 0, 0 \right) \tag{16}$$

and the 3D egomotion parameters are easily computed from it as

$$\alpha_u \frac{T_x}{Z_0} = \frac{p_x}{L_{22}}, \tag{17}$$

$$\alpha_v \frac{T_y}{Z_0} = \frac{p_y}{L_{22}}, \tag{18}$$

$$\frac{T_z}{Z_0} = \frac{1}{L_{22}} - 1. \tag{19}$$

5. Qualitative measure of distance to the target

We propose a qualitative measure of the distance from the robot to the target based on the computation of the time to contact. The time to contact (TTC) is the time needed for the viewer to reach the target if the viewer continues with the same speed. In fact, it is a measure that has been used by different authors for the guidance of wheeled robots [30] or road vehicles [31], assuming motion on a planar surface.

We estimate the likely time to contact with the target by computing the rate of expansion of the target in the image while the camera moves towards it. This calculation can be done without knowledge of either the size and distance of the target, or the speed of the camera towards it.

From Eq. (19) we can observe that the scaled depth of the target can be computed as

$$\frac{Z_0 + T_z}{Z_0} = \frac{1}{L_{22}}.$$

When the viewer goes straight to the target, the time left for collision can be computed from the rate of change in the scaled depth. The system automatically detects this situation from the translation vector estimated using Eqs. (17)–(19). Let H_i be the scaled depth for the contour at frame i

$$H_i = \frac{Z_0 + T_{zi}}{Z_0},$$

where T_{zi} is the translation along Z at frame i.

The difference between H in consecutive frames is

$$H_i - H_{i-1} = \frac{T_{zi} - T_{zi-1}}{Z_0}.$$

Therefore,

$$(H_i - H_{i-1})\frac{1}{H_i} = \frac{T_{zi} - T_{zi-1}}{Z_0 + T_{zi}} = \frac{-1}{\tau},$$

where τ is the time to contact taking the sampling period as time unit.

From this result, we can state that the time to contact can be computed directly from the shape vector as

$$\tau = \frac{H_i}{H_{i-1} - H_i}.$$

The implementation of the theory shows that this measure is a useful tool to predict the collision time. We report an experiment carried out inside a laboratory in order to estimate the reliability of the results. Fig. 2 depicts the situation in which the experiment is set. The sequence was recorded moving the camera at a constant velocity of approximately 16 cm per time unit, and the target was set at 97 cm from the initial position. A simple target was chosen, although the capability of active contours to track complex shapes and their robustness to occlusions have been proved elsewhere [23,26]. Fig. 3 shows the initial image in a sequence taken while the viewer moves towards the target. In this case the target is the black square. Fig. 4 shows four samples of the sequence recorded at time instants one to four as indicated in Fig. 2. The shape vectors for these examples are

$$\begin{aligned} X_A &= [0, 0, 0.219, 0.219, 0, 0], \\ X_B &= [0, 0, 0.563, 0.563, 0, 0], \\ X_C &= [0, 0, 1.196, 1.196, 0, 0], \\ X_D &= [0, 0, 2.690, 2.690, 0, 0]. \end{aligned} \tag{20}$$

Fig. 5 plots the recovered TTC as a function of time. It can be observed that the graphic decreases linearly as

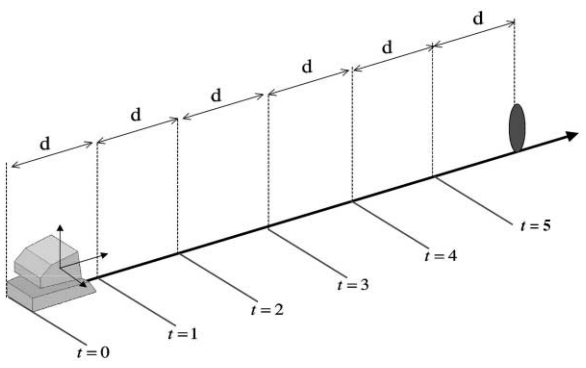

Fig. 2. Experiment to evaluate the TTC computation.

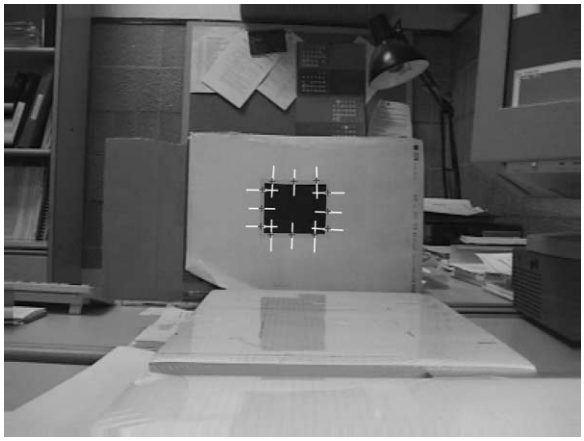

Fig. 3. First image in a sequence recorded to validate the TTC computation.

predicted for a uniform motion. The relative error is under 2%. Note that the affine transformation breaks down when the viewer is very close to the target and, therefore, the estimate of time to contact may be degradated in this case.

6. Computation of the heading direction

The heading direction is represented in the image plane as the point of intersection between this direction and the image plane. It is equivalent to the projection of the translation vector onto the image plane

$$\begin{bmatrix} e_x \\ e_y \end{bmatrix} = \begin{bmatrix} \alpha_u \dfrac{T_x}{T_z} + u_0 \\ \alpha_v \dfrac{T_y}{T_z} + v_0 \end{bmatrix}. \tag{21}$$

From Eqs. (17)–(19) we have

$$\begin{aligned} \alpha_u \frac{T_x}{T_z} &= \frac{p_x}{1 - M_{22}}, \\ \alpha_v \frac{T_y}{T_z} &= \frac{p_y}{1 - M_{22}}, \end{aligned} \tag{22}$$

that lead us to the heading direction if the principal point is known.

The principal point can be self-calibrated by combining the analysis of the active contour with a set of point matches.

A point match $(\mathbf{u}^{(1)}, \mathbf{u}^{(2)})$ fulfils the following equation in homogeneous notation (see Ref. [32] for details)

$$\mathbf{u}^{(2)} = \mathbf{A}\mathbf{R}\mathbf{A}^{-1}\mathbf{u}^{(1)} + \frac{\mathbf{A}\mathbf{T}}{Z_i}, \tag{23}$$

where Z_i is the depth of the 3D point and $\mathbf{A}$ is the calibration matrix. In particular, when the rotation has

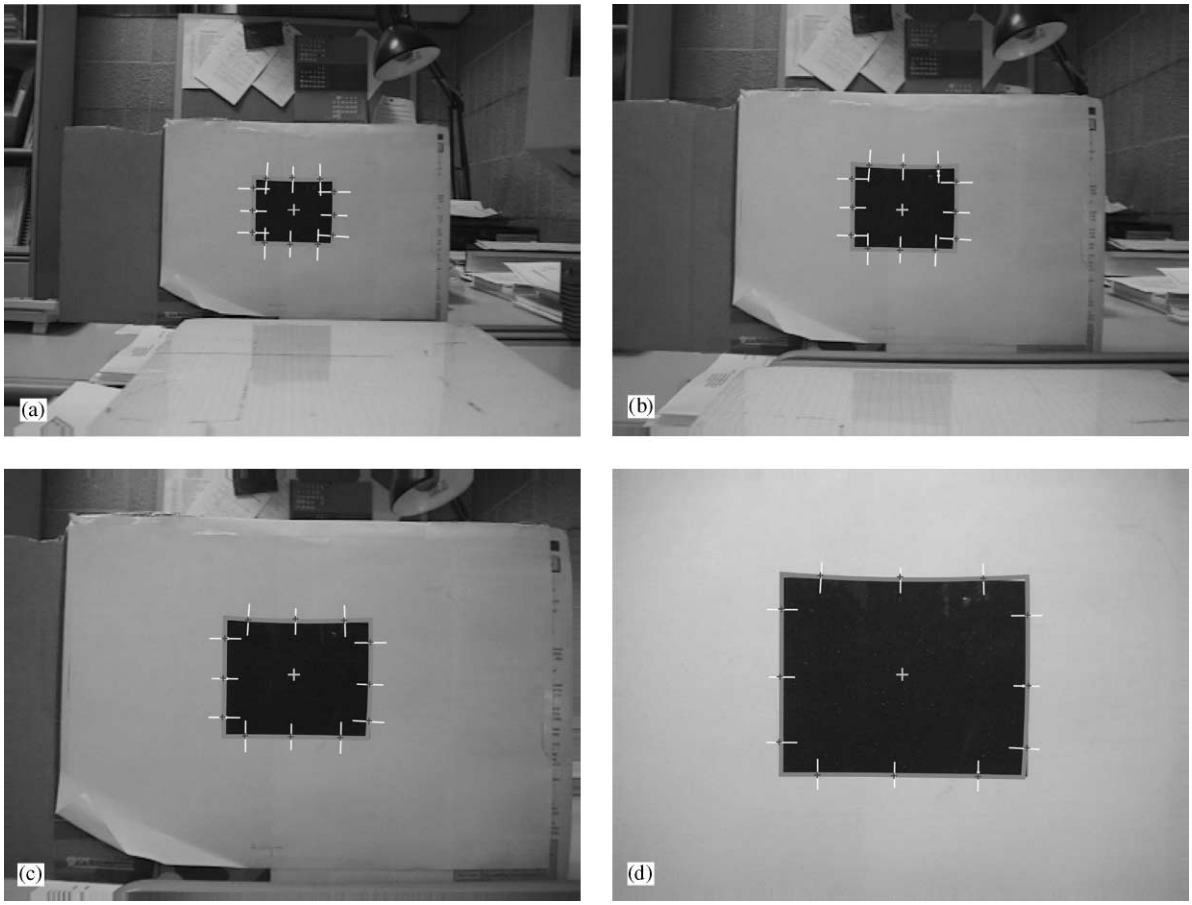

Fig. 4. Estimation of TTC from the deformation of an active contour. Four samples of a video sequence taken by a moving observer approaching the target at a uniform velocity. The images were recorded at time instants one to four as indicated in Fig. 2. The camera had moved 16 cm between consecutive images. An active contour tracks the target. Its deformations are used to estimate the time to contact (Fig. 5). The next image in the sequence corresponds to collision.

been compensated for and the image has only the effects of the translation, the above equation simplifies to

$$\mathbf{u^{(2)}} = \mathbf{u^{(1)}} + \frac{\mathbf{AT}}{Z_i} = \mathbf{u^{(1)}} + \begin{bmatrix} \alpha_u \dfrac{T_x}{T_z} + u_0 \\ \alpha_v \dfrac{T_y}{T_z} + v_0 \\ 1 \end{bmatrix} \frac{T_z}{Z_i}. \tag{24}$$

When the translation along Z is not null it can be rewritten as

$$\mathbf{u^{(2)}} = \mathbf{u^{(1)}} + \begin{bmatrix} e_x \\ e_y \\ 1 \end{bmatrix} \frac{T_z}{Z_i},$$

where (e_x, e_y) are the components of the epipole. Going back to nonhomogeneous notation, we have two linear equations with three unknowns (the components of the epipole and the relative depth Z_i/T_z)

$$u_x^{(2)} = e_x \frac{Z_i}{T_z}(u_x^{(2)} - u_x^{(1)}),$$

$$u_y^{(2)} = e_y \frac{Z_i}{T_z}(u_y^{(2)} - u_y^{(1)}). \tag{25}$$

Each new matching adds two equations and one unknown (the relative depth of the new 3D point). We take a set of point matches and solve for the unknowns by least mean squares. Once the epipole is known, the principal point can be computed from Eqs. (21) and (22),

$$u_0 = e_x - \alpha_u \frac{T_x}{T_z},$$

$$v_0 = e_y - \alpha_v \frac{T_y}{T_z}.$$

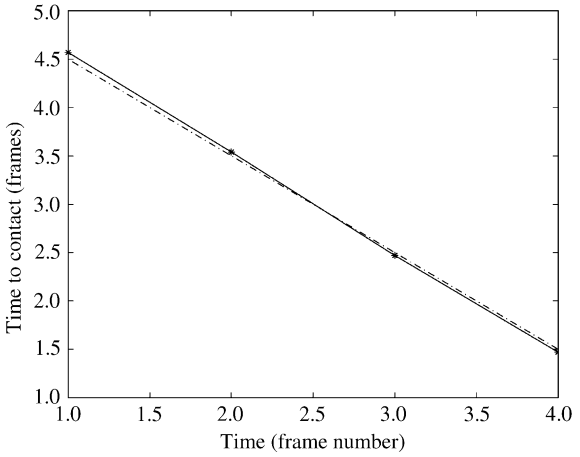

Fig. 5. Estimated time to contact. The solid line depicts the estimated time to contact, while the dotted one is the expected time to contact, which decreases linearly for a uniform motion. It can be observed that the error decreases as the camera approaches the target until the viewer is very close to the target, then the affine transformation breaks down and the estimate is degradated.

If there is no translation along Z, Eq. (25) reduces to

$$u_x^{(2)} = u_x^{(1)} + \alpha_u \frac{T_x}{Z_i},$$

$$u_y^{(2)} = u_y^{(1)} + \alpha_v \frac{T_y}{Z_i}. \tag{26}$$

In this case the principal point cannot be computed. The analysis of contour deformations allows the system to detect when the translation along Z is null. Only when it is not null, the system extracts the principal point.

The principal point often remains constant for long periods of time, during which the analysis of the contour provides enough information to guide the matching between frames. Only when the principal point changes (i.e. when the camera zooms) the contour has to be combined with point matches to recalibrate the camera.

7. Matching between frames. Computation of epipolar lines

In this section we explain how the epipolar geometry can be deduced from the analysis of an active contour. The epipolar geometry is the only relation we can obtain that describes the matching between two uncalibrated images. We are interested in matching different points of two views of the same scene in order to recover the depth of these points. Once the depth of a set of points is known, it can be interpolated to obtain an approximate depth map of the whole scene.

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. It is important to switch to a full-perspective camera model because we are interested in extracting the epipolar lines corresponding to points in the image that may be at different distances. The affine camera is adequate to model the imaging process of the target, as it is assumed that the target occupies a small region in the image and its depth range is small compared to its distance to the camera. However, this simplified model 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 $\mathbf{u}^{(1)}$. Therefore, its corresponding point in the second image, $\mathbf{u}^{(2)}$, should lie on the projection of this ray, namely, the epipolar line of $\mathbf{u}^{(1)}$. The epipolar lines simplify the correspondence problem because the search for matches is reduced to a 1D search. All epipolar lines intersect at the projection of the optical centre of the camera at its first location in the other camera location, namely the epipole.

The epipolar lines are usually computed from the fundamental matrix $\mathbf{F}$, which is a 3×3 matrix that describes the correspondence between two images of the same scene recorded from different viewpoints [33–35]. It relates the projections $\mathbf{u}^{(1)}$, $\mathbf{u}^{(2)}$ of a 3D point, in homogeneous notation as follows:

$$\mathbf{u}^{(2)\mathbf{T}}\mathbf{F}\mathbf{u}^{(1)} = 0. \tag{27}$$

$\mathbf{F}$ can be split up [33,35,36] as

$$\mathbf{F} = \mathbf{A}^{-\mathbf{T}}[\mathbf{T}]_* \mathbf{R}\mathbf{A}^{-1},$$

where $\mathbf{A}$ is the calibration matrix,

$$\mathbf{A} = \begin{bmatrix} \alpha_u & 0 & u_0 \\ 0 & \alpha_v & v_0 \\ 0 & 0 & 1 \end{bmatrix},$$

$\mathbf{A}^{-\mathbf{T}}$ is the transpose of $\mathbf{A}^{-1}$ and $[\mathbf{T}]_*$ is a matrix obtained from the elements of $\mathbf{T}$,

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

When the rotation has been compensated for, the epipoles are the same for both images and equal the heading direction. In this case, $\mathbf{F}$ simplifies to

$$\mathbf{F} = \mathbf{A}^{-\mathbf{T}}[\mathbf{T}]_*\mathbf{A}^{-1} = [\mathbf{A}\mathbf{T}]_* = [\mathbf{e}^{(2)}]_* = [\mathbf{e}^{(1)}]_*. \tag{28}$$

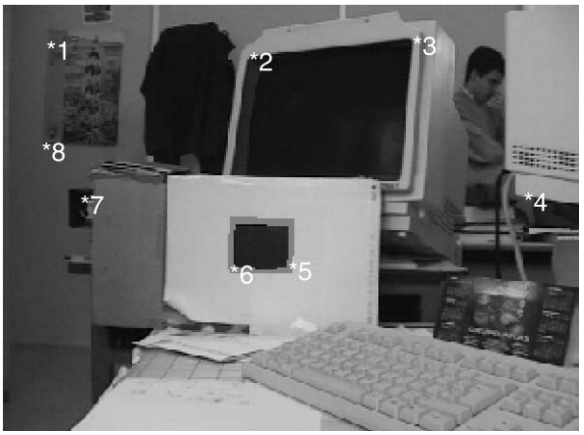

Fig. 6. The square is taken as target in the initial image. The deformation of the contour fitted to it allows us to recover the heading direction and epipolar lines. The latter will be computed for the indicated points, as an example.

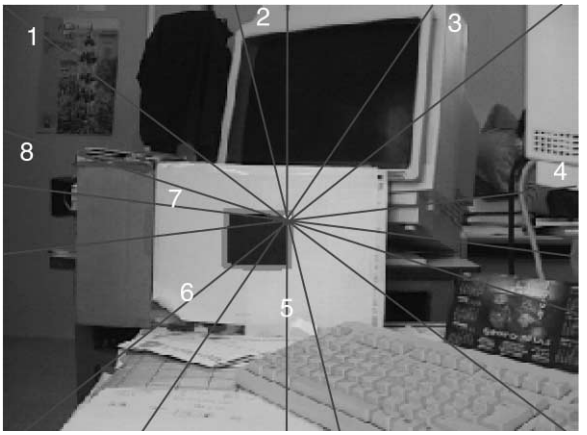

Fig. 7. Epipolar lines relating the image in Fig. 6 with the image recorded after a translation along the Z direction. The heading direction points towards the epipole, and the epipole is the point of intersection of all epipolar lines.

Using homogeneous notation, a line $\mathbf{l}^{(2)}$ passing through a point $\mathbf{u}^{(2)}$ fulfils the following equation (an introduction to perspective geometry can be found in Ref. [32]):

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

Therefore, from Eq. (27), the epipolar line can be computed as

$$\mathbf{l}^{(2)} = \mathbf{F}\mathbf{u}^{(1)} = \mathbf{A}^{-\mathrm{T}}[\mathbf{T}]_{*}\mathbf{A}^{-1}\mathbf{u}^{(1)} = [\mathbf{AT}]_{*}\mathbf{u}^{(1)}.$$

From Eq. (28),

$$\mathbf{l}^{(2)} = [\mathbf{e}^{(2)}]_{*}\mathbf{u}^{(1)}.$$

The epipolar line coincides with the line $\mathbf{l}$ joining the epipole with $\mathbf{u}^{(1)}$

$$\mathbf{l} = \mathbf{e}^{(2)} \times \mathbf{u}^{(1)} = (\mathbf{AT}) \times \mathbf{u}^{(1)} = [\mathbf{AT}]_{*}\mathbf{u}^{(1)}.$$

Thus, the computation of the epipole allows us to draw the epipolar lines. Some results are shown in Figs. 6–8. If the disparity between images increases, an algorithm based exclusively on point matches would fail, as it would not be able to find reliable matches. On the contrary, the method based on contours maintains a right measure of the epipolar geometry. Moreover, it confines the search for a match to a segment of the epipolar line. If the camera is approaching the target the search should proceed from the point in the first image to the epipole, while it should proceed away from the epipole when the camera recedes from the target. When there is no translation along Z, the search should follow the same direction as the target.

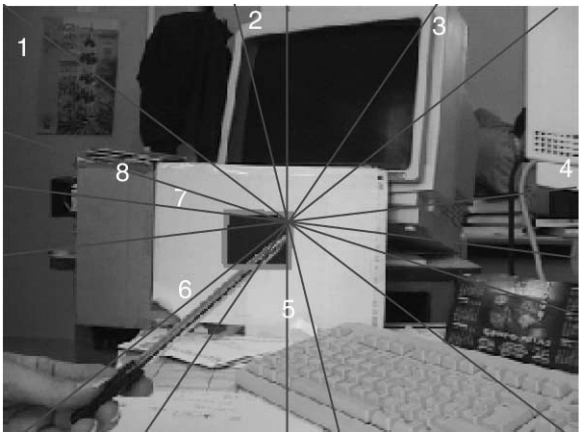

Fig. 8. The epipolar lines are right even when there are either independent motions in the scene or occlusions of the target.

8. Qualitative 3D scene reconstruction

The 3D structure of the visible environment can be specified by the distance along the optical axis (the depth) of each point in the image. Some applications may require a description of solid shapes. In this case, there must also be a transformation from the pointwise description to a solid shape. However, in this section we restrict our attention to pointwise 3D information and we interpolate the result to obtain an approximation of the structure of the whole scene assuming that the surface is smooth.

8.1. Proposed scheme

Once the principal point is known, the epipole can be extracted from the analysis of the contour. We use it not

only to know the heading direction, but also to draw the epipolar lines. Thus, matches between frames are more easily found. Once point matches are achieved, from Eqs. (19) and (25), we can solve for the scaled depth,

$$\frac{Z_i}{Z_0} = \frac{(e_x - u_x^{(2)})T_z/Z_0}{u_x^{(2)} - u_x^{(1)}} = \frac{(e_x - u_x^{(2)})1/L_{22}}{u_x^{(2)} - u_x^{(1)}},$$

$$\frac{Z_i}{Z_0} = \frac{(e_y - u_y^{(2)})T_z/Z_0}{u_y^{(2)} - u_y^{(1)}} = \frac{(e_y - u_y^{(2)})1/L_{22}}{u_y^{(2)} - u_y^{(1)}}.$$

The epipole and T_z/Z_0 are the same for all points in the same frame. The magnitude of the depth of a point is

$$\left|\frac{Z_i}{Z_0}\right| = \left|\frac{T_z}{Z_0}\right| \frac{|\mathbf{e} - \mathbf{u^{(2)}}|}{|\mathbf{u^{(2)}} - \mathbf{u^{(1)}}|},$$

where **e** is the epipole.

The above deduction is valid as far as there is a non-null translation along Z. The analysis of contour deformations allows one to detect when the translation along Z is null and use a different set of equations to solve for 3D scene structure in this case. When there is no translation along Z, the scaled depth of the points can be recovered from the combination of the scaled translation obtained from the active contour and the matching of points using Eq. (26), which is repeated here for convenience,

$$u_x^{(2)} = u_x^{(1)} + \alpha_u \frac{T_x}{Z_i},$$

$$u_y^{(2)} = u_y^{(1)} + \alpha_v \frac{T_y}{Z_i}.$$

Using point matches, we can compute the value of $\alpha_u(T_x/Z_i)$, $\alpha_v(T_y/Z_i)$. If we combine it with the scaled translation $(\alpha_u(T_x/Z_0), \alpha_v(T_y/Z_0))$ obtained from the analysis of the contour, we get the scaled depth Z_i/Z_0.

Regardless of whether there is a translation along Z, the depth map is improved by adding the points inside the target to the set of points for which the depth is known. From Eq. (19) we have an approximation of the depth of points inside the target

$$\frac{Z_0 + T_z}{Z_0} = \frac{1}{L_{22}}.$$

Fig. 9 depicts the proposed scheme. We emphasise the fact that there is an initialization step, in which the principal point is computed. After this, a very simple scheme makes it possible to extract both the epipolar geometry and the 3D structure. Once the process has been initialized, the epipolar geometry is directly extracted from the deformations of the contour.

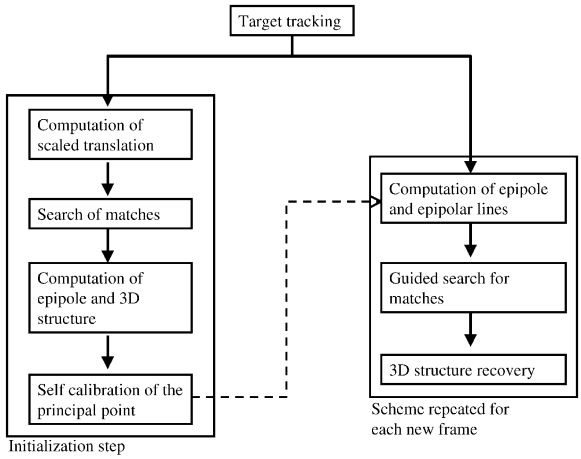

Fig. 9. Proposed scheme to recover epipolar geometry and 3D structure.

8.2. Experimental results

The proposed algorithm has been tested on several image sequences, and good results have been obtained. At a first stage the algorithm was evaluated using indoor scenes, but later it has been successfully applied to real outdoor scenes. Here we provide the qualitative depth map of one of these scenes. Figs. 10 and 11 show different frames of the scene with a translation between them. An active contour is fitted to the target, which is the door.

Fig. 12 highlights some points in the image, for which the epipolar lines are drawn in Fig. 13. A set of salient features are automatically detected in Fig. 14. Among the available ways to detect salient points in an image [37,38], we have used the technique proposed by Zhang [39]. The points are matched with the ones in Fig. 11 using the epipolar lines computed from the analysis of the active contour. Once the matches are known, the depth map is computed. Fig. 15 shows the result. Blue colours depict distant regions while red ones represent near points. For instance, the silhouette of the tree can be observed, in red, in the left side of the image. Few points are detected on the house, therefore the 3D structure recovered is not very accurate in this zone. This result is improved when the reconstruction is enriched by adding the estimated depth of the points inside the target to the depth of the matched salient points. Fig. 16 depicts a view of the final result. We can observe the shape of the tree in red, the target and background points in blue, the grass in a combination of yellows, which means that it is in between the tree and the target. In front of the target, we can observe the shape of a branch of the tree and the reconstruction of points that belong to the tractor. All these data about the environment structure are a

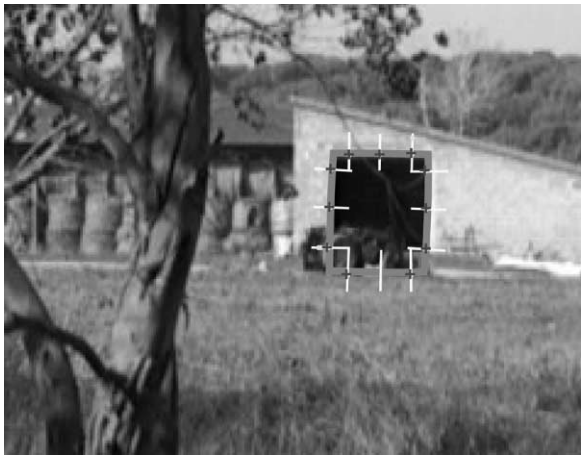

Fig. 10. An active contour is fitted to the target.

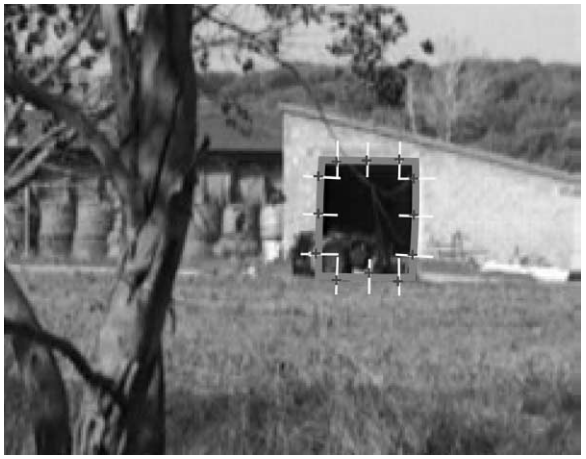

Fig. 11. The target is tracked along the sequence.

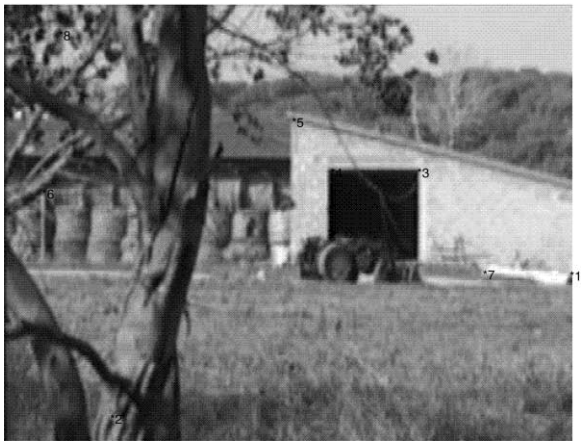

Fig. 12. The epipolar lines will be computed for the set of salient points drawn in this figure.

Fig. 13. The epipolar lines corresponding to the image points in Fig. 12 are drawn.

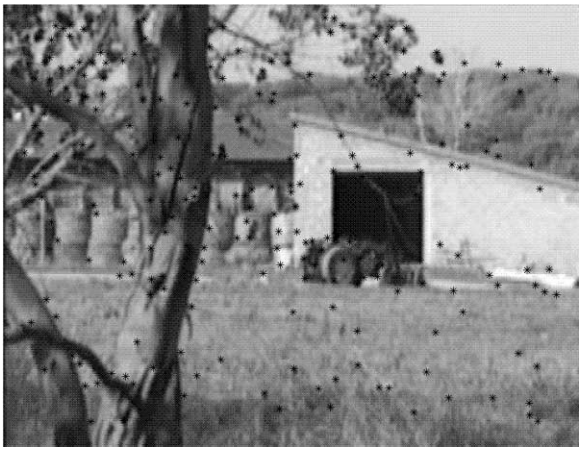

Fig. 14. Whole set of salient points of the image. Matches for them between frames are found using the epipolar lines computed from the deformation of the active contour.

valuable information for the robot to decide its best path towards the target.

8.3. Error analysis and estimation

In our approach, uncertainty arises from several sources:

- orientation compensation,
- edge detection,
- active contour fitting,
- detection of salient features,
- matching between frames.

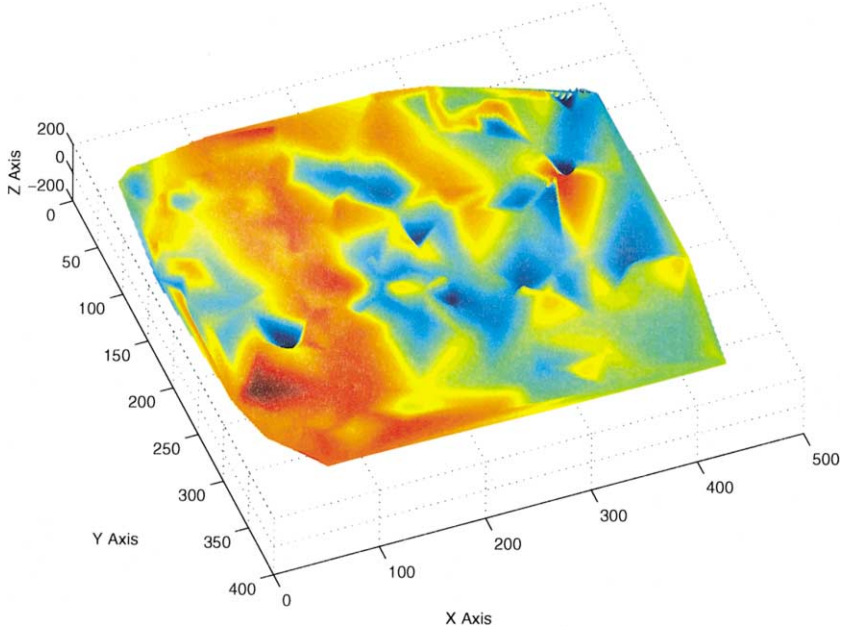

Fig. 15. 3D structure recovery from interpolation of the depth of matched points.

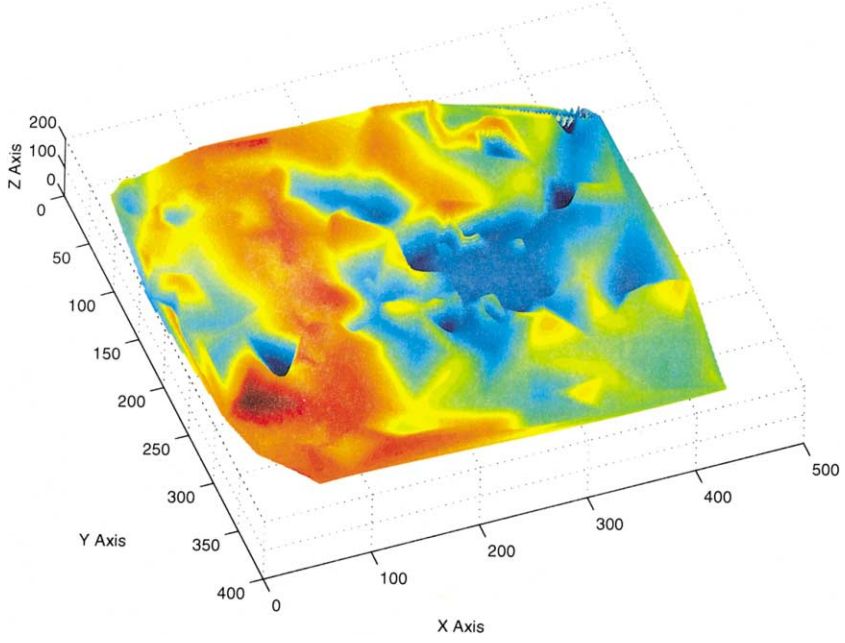

Fig. 16. 3D structure recovery adding the depth of the points inside the target. This reconstruction has been computed by interpolating the depth of the matched points and that of the points inside the target.

They introduce errors in the motion estimate and in the 3D depth map.

An error in the compensation of rotation causes a wrong interpretation of the affine deformation in the image plane. We use Eqs. (10) and (11), when the correct ones in this case are Eqs. (6) and (7). This introduces an error in the computation of the scaled translation which is transferred to the epipole estimate. A perturbation in

the epipole displaces the epipolar lines. If the perturbation is small, the epipolar lines can still be used by searching for matches in a region around the estimated epipolar line. Obviously, the efficiency of the method decreases as the perturbation in the compensation of rotation increases.

On the other hand, the equations we use to compute the depth assume that the rotation has been completely compensated (Eq. (24)). Using these equations when there is rotation introduces an error term ε such that

$$\mathbf{u^{(2)}} - \mathbf{u^{(1)}} = \frac{\mathbf{AT}}{Z_i} + \varepsilon,$$

where, from Eqs. (23) and (24),

$$\varepsilon = (\mathbf{ARA^{-1}} - \mathbf{I})\mathbf{u^{(1)}}.$$

Note that the error term does not only depend on the rotation matrix, but also on the calibration matrix. When the rotation is not completely compensated, the scaled depth of points (Z_i/T_z) can only be computed if the internal calibration parameters of the camera are known.

Apart from the compensation of rotation, the other error sources are associated to measurements. Errors may arise in the extraction of salient features and their matching between frames, as well as in edge detection and the subsequent contour fitting. Weng et al. [40] address this kind of errors and derive how they affect to the 3D structure and motion estimation from point matches. Here we focus on the propagation of the error related to the use of an active contour as the basis to compute the 3D motion.

The Kalman filter [41] provides an estimate $\bar{X}$ of the shape vector of the target's contour at each frame. It yields only an estimate of the shape vector, but also an estimate of its covariance matrix. Once the shape vector $\mathbf{X}$ is estimated and its covariance matrix $\Gamma_{\mathbf{X}}$ is computed, we extract the motion parameters. We analyse how the uncertainty in the shape vector propagates to the motion parameters and the scaled depth of the contour. Assuming a small perturbation in the original data, we analyse the linear terms of the perturbation of the final result to estimate the errors.

The Kalman filter provides the estimation $\bar{X}$ of the shape vector and the covariance matrix $\Gamma_{\mathbf{X}}$ assuming that the shape vector follows a gaussian distribution. From the third component $(\bar{X}_3)$ of the shape vector estimate we have an estimation of the scaled depth (see Eq. (16)),

$$H \triangleq \frac{Z_0 + T_z}{Z_0} = \frac{1}{X_3 + 1}.$$

We approximate H up to the linear term of a Taylor series expansion about $X_3 = \bar{X}_3$, leading to an approximation of the covariance of H (Γ_H) as a function of the covariance of X_3 (Γ_{X_3}),

$$\Gamma_H = \frac{1}{(\bar{X}_3 + 1)^4}\Gamma_{X_3}, \tag{29}$$

which can be approximated as

$$\Gamma_H = \bar{H}^4\Gamma_{X_3},$$

where $\bar{H}$ is the mean of H. When the camera moves towards the target ($H \in (0,1]$) the covariance of the scaled depth is smaller than the one for the third component of the shape vector ($\Gamma_H < \Gamma_{X_3}$). In this case, the propagated error is bounded. However, when the viewer moves away from the target the covariance of the scaled depth increases with exponent four and the uncertainty is unbounded.

9. Concluding remarks

This paper presents a new approach to provide a walking robot with qualitative information to reach a visual target. The work highlights the benefits of using the information derived from an active contour to guide the matching of features between frames. The proposed method is based on a direct measure of image deformation from an active contour fitted to a target. It is essentially different from the common techniques that use velocity or displacement fields as the unique basis for further computation [19,42–44].

Several advantages are attained by focussing the processing on the target. The first one is speed, the epipolar geometry is recovered at frame rate from live video (25 frames/s) using a Silicon Graphics Indy at 150 MHz. The size of the images is 485×372, but note that the analysis of the contour is invariant to the image dimensions as it is carried out in a local region. The second advantage is the robustness of the method to independent motions in the scene. It is remarkable to observe that most of the current methods to compute the epipolar geometry rely on the assumption of a single independent, motion; i.e., they work for scenes containing only one moving object or, alternatively, a moving camera in a stationary environment [35,45]. The third one is that the focusing mechanism allows us to assume a simplified camera model for points in the target, no matter if this model does not fit the rest of the image. The proposed method relies on a combination of an affine camera and a full-perspective camera. Once the motion parameters have been recovered using the simplified camera model, the epipolar geometry and scene structure are computed using a full-perspective camera model. Therefore, we combine the generality of a full-perspective camera model with the robustness of a scheme based on linear approximations.

The method is limited to situations in which the target is static and visible under a weak perspective assumption. In this case, the usual techniques based only on point matches can be complemented with the information from the contour.

The traditional approach requires an initial set of reliable matches to extract the epipolar geometry and then guide the search for additional matches [33–35], while the proposed method takes advantage of the analysis of the contour to avoid the initial unguided matching. Moreover, a number of the previous works rely on the computation of the fundamental matrix, which becomes unstable when the matched points are coplanar. In this situation, it is better to describe the relation between two views by a homography instead of a fundamental matrix [32,33]. The key question is to know when to switch from using the fundamental matrix to using a homography, and vice versa. The proposed method is invariant to the distribution of salient points in the image. The epipolar geometry is recovered directly from the active contour; therefore it does not become unstable when salient points are coplanar. In addition, the analysis of the active contour allows one to estimate a qualitative measure of depth, namely the time to contact, even when there are no salient points in the scene. The traditional approach based only on matched points limits the extraction of 3D information to those scenes in which a set of salient points can be detected.

Future work will include a more thorough analysis of error propagation and estimation, as well as the extension of the method to using several contours fitted to different regions in the image. The fusion of the information provided by different contours would make the process more robust. Moreover, once a contour is fitted to a region, we have proved that its scaled depth can be computed and used to enrich the 3D reconstruction of the whole scene. The final depth map is expected to combine the depth of point features with the depth of the contours fitted to homogeneous regions. Thus, by exploiting the information provided by the active contours, the extraction of depth information is not limited to those scenes in which salient points can be detected.

Acknowledgements

The authors wish to thank Andrew Blake for very useful discussions and guidance in the first stages of this work. The research has been partially supported by the research grant "Navegación basada en visión de robots autónomos en entornos no estructurados" CICYT TAP97-1209 of the Spanish Science and Technology Council, and the grant 1997BEAI200071 of the Direcció General de Recerca of the Generalitat of Catalunya.

References

[1] E. Celaya, F. Porta, Control of a six-legged robot walking on abrupt terrain, Proceedings of IEEE International Conference on Robotics and Automation, Minneapolis, MN, 1996, pp. 2731–2736.

[2] E. Celaya, F. Porta, A control structure for the locomotion of a legged robot on difficult terrain, IEEE Robotics Automat. Mag. 5 (1998) 43–51.

[3] S. Soatto, P. Perona, Recursive estimation of camera motion from uncalibrated image sequences, Proceedings of the First IEEE International Conference on Image Processing (ICIP), 1994, pp. II-58–62.

[4] J.L. Crowley, P. Bobet, C. Schmidt, Maintaining stereo calibration by tracking image points, Proceedings of the Conference on Computer Vision and Pattern Recognition, 1993, pp. 483–488.

[5] I.A. Lourakis, R. Deriche, Camera self-calibration using the singular value decomposition of the fundamental matrix: from point correspondences to 3d measurements, Technical Report, No. 3748. INRIA, Sophia Antipolis, France, 1999.

[6] C. Zeller, O. Faugeras, Applications of non-metric vision to some visual guided tasks, Technical Report, No. 2308. INRIA, Sophia Antipolis, France, 1994.

[7] E. Martínez, C. Torras, Integration of appearance and geometric methods for the analysis of monocular sequences, Proceedings of the IST/SPIE 12th Annual Symposium on Electronic Imaging, San Jose, California, January 2000, pp. 62–70.

[8] A. Kosaka, J. Pan, Purdue experiments in model based vision for hallway navigation, Proceedings of Workshop on Vision for Robots in IROS'95 Conference, 1995, pp. 87–96.

[9] C.J. Taylor, D.J. Kriegman, Vision based motion planning and exploration algorithms for mobile robots, IEEE Trans. Robotics and Automation 14 (3) (1998) 417–426.

[10] E.D. Dickmanns, B. Mysliwetz, T. Christians, An integrated spatio-temporal approach to automatic visual guidance of autonomous vehicles, IEEE Trans. Systems Man Cybernet. 20 (6) (1990) 1273–1284.

[11] E. Gat, F. Desai, R. Iulev, J. Loch, D.P. Milter, Behaviour control for robotic exploration of planetary surfaces, IEEE Trans. Robotics Automat. 10 (4) (1994) 490–503.

[12] A. Davison, D.W. Murray, Mobile robot localisation using active vision, Proceedings of the Fifth European Conference on Computer Vision, 1998.

[13] J. Navarrete, J. Alberdi, J. Barcala, V. Chuatchkine, E. Gamero, I. Ioudine, A. Molinero, C. Yuste, Sistema de vision tridimensional del robot movil de locomocion sobre patas rimho. Proceedings of IV Congreso espanol de la AER, Zaragoza, Spain, October 1995, pp. 85–91.

[14] D. Pack, Perceptual-based control for a quadruped walking robot, Proceedings of the IEEE International Conference on Robotics and Automation, 1996, pp. 2994–3001.

[15] O. Faugeras, F. Lustman, G. Toscani, Motion and structure from motion from point and line matches, Proceedings of the First International Conference on Computer Vision, 1987, pp. 25–34.

[16] R. Cipolla, Y. Okamoto, Y. Kuno, Robust structure from motion using motion parallax, Fourth International Conference on Computer Vision (ICCV93), 1993, pp. 374–382.

[17] P.A. Beardsley, A. Zisserman, D.W. Murray, Sequential updating of projective and affine structure from motion, Int. J. Comput. Vision 23 (3) (1997) 235–259.
[18] T.J. Broida, S. Chandrashekhar, R. Chellappa, Recursive estimation of 3-d kinematics and structure from a noisy monocular image sequence, Proceedings of ISCV, November 1995.
[19] T.Y. Tian, C. Tomasi, D.J. Heeger, Comparison of approaches to egomotion computation, Proceedings of the Conference on Computer Vision and Pattern Recognition, 1996, pp. 315–320.
[20] Z. Zhang, A new multistage approach to motion and structure estimation: from essential parameters to euclidean motion via fundamental matrix, Technical Report No. 2910, INRIA, Sophia Antipolis, France, 1996.
[21] P.F. McLauchlan, D.W. Murray, A unifying framework for structure and motion recovery from image sequences, Proceedings of the Fifth International Conference on Computer Vision, Cambridge, MA, 1995, pp. 314–320.
[22] R. Cipolla, A. Blake, Surface orientation and time to contact from image divergence and deformation, Proceedings of the Second European Conference on Computer Vision, 1992, pp. 187–202.
[23] A. Blake, M. Isard, Active Contours, Springer, Berlin, 1998.
[24] J.L. Mundy, A. Zisserman, Geometric Invariance in Computer Vision, MIT Press, Cambridge, MA, 1992.
[25] J. Foley, A. van Dam, S. Feiner, F. Hughes, Computer Graphics, Principles and Practice, Addison-Wesley, Reading, MA, 1996.
[26] A. Blake, M.A. Isard, D. Reynard, Learning to track the visual motion of contours, J. Artif. Intell. 78 (1995) 101–134.
[27] C. Fermuller, Y. Aloimonos, Qualitative egomotion, Int. J. Comput. Vision 15 (1995) 7–29.
[28] C. Tomasi, J. Shi, Direction of heading from image deformations, IEEE Conference on Computer Vision and Pattern Recognition, 1993, pp. 422–427.
[29] J. Lawn, R. Cipolla, Epipole estimation using affine motion parallax, Proceedings of the British Machine Vision Conference, 1993.
[30] J. Santos-Victor, G. Sandini, Visual behaviors for docking, Technical Report, LIRA-Lab-DIST University of Genova, TR 2/94, 1994.
[31] E.D. Dickmanns, V. Graefe, Dynamic monocular machine vision, Mach. Vision Appl. 1 (1988) 223–240.
[32] O. Faugeras, 3D Computer Vision, MIT Press, Cambridge, MA, 1993.
[33] Q.T. Luong, R. Deriche, O. Faugeras, T. Papadopoulo, On determining the fundamental matrix: analysis of different methods and experimental results, Technical Report, INRIA, Sophia Antipolis, RR 1894, 1993.
[34] P.H.S. Torr, D.W. Murray, The development and comparison of robust methods for estimating the fundamental matrix, International Journal on Computer Vision 24 (3) (1997) 271–300.
[35] Z. Zhang, Determining the epipolar geometry and its uncertainty: a review, Technical Report, No. 2927, INRIA, Sophia Antipolis, France, 1996.
[36] R. Deriche, Z. Zhang, Q.T. Luong, O. Faugeras, Robust recovery of the epipolar geometry for an uncalibrated stereo rig. Proceedings of the Third European Conference on Computer Vision, Stockholm, 1994, pp. 567–576.
[37] J. Shi, C. Tomasi, Good features to track, Proceedings of the Conference on Computer Vision and Pattern Recognition, 1994, pp. 593–600.
[38] L. Shapiro, Affine Analysis of Image Sequences, Cambridge University Press, Cambridge, 1995.
[39] Z. Zhang, R. Deriche, O. Faugeras, Q.T. Luong, A robust technique for matching two uncalibrated images through the recovery of the unknown epipolar geometry, Artif. Intell. J. 78 (1995) 87–119.
[40] J. Weng, T.S. Huang, N. Ahuja, Motion and Structure from Image Sequences, Springer, Berlin, 1993.
[41] S.M. Bozic, Digital and Kalman Filtering, An Introduction to Discrete-Time Filtering and Optimum Linear Estimation, Edward Arnold, London, 1979.
[42] C. Harris, Determination of egomotion from matched points, Proceedings of the Third Alvey Vision Conference, 1987.
[43] C. Silva, J. Santos-Victor, Robust egomotion estimation from the normal flow using search subspaces, IEEE Trans. Pattern Anal. Mach. Intell. 19 (9) (1997) 1026–1034.
[44] A. Verri, E. Trucco, Finding the epipole from uncalibrated optical flow, Proceedings of British Machine Vision Conference, 1997, pp. 605–609.
[45] M. Irani, B. Rousso, S. Peleg, Recovery of egomotion using image stabilization, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Seatle, WA, June 1994, pp. 454–460.

About the Author—ELISA MARTÍNEZ MARROQUÍN received her B.Sc. degree in Telecommunication Engineering and her M.Sc. degree in Electrical Engineering from Ramon Llull University, Barcelona, Spain, in 1993 and 1995, respectively. In 1996 she became an assistant professor in the Communications and Signal Theory Department, Ramon Llull University, where she is currently concluding her Ph.D. thesis. Her current research interests are in the area of computer vision, including structure from motion, egomotion estimation, object tracking and visual guidance of mobile robots.

About the Author—CARME TORRAS GENÍS received her M.Sc. degree in Mathematics from the Universitat de Barcelona in 1978, her M.Sc. degree in Computer Science from the University of Massachusetts at Amherst in 1981, and her Ph.D. degree in Computer Science from the Universitat Politècnica de Catalunya in 1984. She holds a position of Professor of Research in the Consejo Superior de Investigaciones Científicas (CSIC) and her major research interests are Robot Motion Planning and Neurocomputing.