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. Email: elisa@salleurl.edu http://www.salleurl.edu/˜elisa*

[b] *Institut de Robòtica i Informàtica Industrial. Gran Capità, 2-4, 08034 Barcelona, Spain. Email: ctorras@iri.upc.es http://www- iri.upc.es/people/torras*

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 autofocus, freely moving towards a stationary target in an unstructured environment that may contain independently moving objects. The goal is to dynamically analyse the sequence in order to extract information about the robot motion, the target position and the environment structure. The deformations of an active contour fitted to the target "permits recovering" the scaled egomotion and time-to-contact with the target, at frame rate. From which, together with a procedure for the self-calibration of the principal point, the epipolar lines can be readily derived. These lines speed up drastically the matching of salient points needed to recover 3D structure, by reducing the search process from 2D to 1D. The 3D reconstruction is performed using a full perspective camera model, whereas an affine model sufficed for all the previous stages. 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.

Keywords: Sensory-motor integration for visual tasks, contour tracking, qualitative navigation, egomotion, time to contact, heading direction, qualitative 3D reconstruction of image sequences.

1 Introduction

The present work is part of a project aimed at the development of a low-cost walking robot for exploratory tasks [5,6]. 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.

Note that 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
- compass information
- unstructured environment
- unknown camera motion
- uncalibrated camera
- active camera with pan and autofocus
- 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 [35], others (e.g., image centre and focal length) may change drastically along an image sequence [10]. 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 autocalibration of the camera on-line [26] or in designing methods which do not need the calibration parameters [42]. Among the latter, 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 [28].

In this paper, we describe a technique that provides a qualitative estimation of robot motion, target position and environment structure. Other visual processes related to landmark detection and recognition are described elsewhere [38], as are the aspects related to locomotion and navigation within the project (ref.?).

The procedure 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 method combines the analysis of active contours [2] with the geometric constraints between different views of a single scene, namely the epipolar geometry. An active contour is automatically fitted

Fig. 1. **Global scheme.**

to the target marked by the operator in the image, from which a shape vector is extracted for each frame. The solution is based on a direct measure of image deformation from the analysis of the active contour fitted to the target, which is assumed to be static.

We describe an analysis of the active contour that allows 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. In order to extract information about the scene structure, we propose a solution based on the combination of the analysis of the active contour with the data provided by point matches between the different frames. We prove that this combination allows 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, we explain how 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 interpolating the depth of the matched points and the depth of the points inside the target. The proposed scheme is summarized in Fig. 1.

There are many works dealing with the visual guidance of robots in structured scenes [24,21,36,9]; less works address the visual guidance of wheeled or tracked robots in slightly structured or non structured environments [14,20,18,11]; but works tackling the visual guidance of walking robots in unstructured scenes are very scarce [31,32].

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 [16,8,1,4,37,44,29]. The usual approach to estimate camera motion and scene structure is based on optic flow. This can be computed in two ways, either by obtaning 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 its use may be justified when the structure of the scene needs to be recovered with high accuracy, but this is not the case in our application.

There are a few works that compute egomotion on the basis of only local information. Cipolla and Blake (1992) [7] 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 just 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" [6], a thing that tracked robots cannot do without extra degrees of freedom in the camera subsystem.

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 combination of matches with the analysis of the shape vector allows to self-calibrate the principal point of the camera as described in Section 6. Subsequent computation of the heading direction or epipole is explained in Section 7. Section 8 is devoted to the recovery of the epipolar geometry, which is used to match differents views. The combination of the analysis of the contour with the matches allows to recover a qualitative depth map (Section 9). Finally, the advantages and limitations of the proposed procedure are discussed in Section 10.

2 Projection of 3D motion on 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 the depth variation of its points is small compared to their distances 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

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

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 [30], 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, and the projected curve on the image plane has the following expression,

$$\mathbf{d}(s) = \begin{bmatrix} K_u & 0 \\ 0 & -K_v \end{bmatrix} \frac{f}{\mathbf{R_3 D_0}(s)} \left(\begin{bmatrix} R_{11} & R_{12} & R_{13} \\ R_{21} & R_{22} & R_{23} \end{bmatrix} \begin{bmatrix} X_0(s) \\ Y_0(s) \\ Z_0 \end{bmatrix} + \begin{bmatrix} T_x \\ T_y \end{bmatrix} \right) + \begin{bmatrix} u_0 \\ v_0 \end{bmatrix} \tag{2}$$

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 $\mathbf{T} = (T_x, T_y, T_z)^T$. 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 in different measures. We will finally prove that we can provide the robot with enough information for navigation without apriori knowledge of the calibration parameters.

Without loss of generality, we can assume that the centre of $D_0(s)$ has $X = Y = 0$ components; it is equivalent to assume 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 equation (2) can be rewritten as

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

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 equations (3) and (4),

$$\mathbf{d}(s) - \begin{bmatrix} u_0 \\ v_0 \end{bmatrix} = \frac{Z_0}{R_{33}Z_0 + T_z} \begin{bmatrix} K_u & 0 \\ 0 & -K_v \end{bmatrix} \begin{bmatrix} R_{11} & R_{12} \\ R_{21} & R_{22} \end{bmatrix} \begin{bmatrix} \frac{1}{K_u} & 0 \\ 0 & \frac{-1}{K_v} \end{bmatrix} \left(\mathbf{d_0}(s) - \begin{bmatrix} u_0 \\ v_0 \end{bmatrix} \right) +$$
$$+ \frac{f}{R_{33}Z_0 + T_z} \begin{bmatrix} K_u & 0 \\ 0 & -K_v \end{bmatrix} \left(Z_0 \begin{bmatrix} R_{13} \\ R_{23} \end{bmatrix} + \begin{bmatrix} T_x \\ T_y \end{bmatrix} \right)$$

Now, it is interesting to observe that $\left(\mathbf{d_0}(s) - \begin{bmatrix} u_0 \\ v_0 \end{bmatrix} \right)$ is the template centered 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 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 (equation (1)) projects as an affine deformation of the template onto the image plane (equation (5)), when the curve is viewed underweak 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, which is common in Computer Graphics [19], $\mathbf{d}(s) = (d_x(s), d_y(s))^T$, where both $d_x(s)$ and $d_y(s)$ are B-spline curves. We can write them as a function of their control points,

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

where $\mathbf{Q^i}$ is a column vector of control points for the i-th component and $\mathbf{B}(s)$ is a row vector of B-spline basis functions [19,2].

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 in equation (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 equation (8), the difference between $\mathbf{d}(s)$ and $\mathbf{d_0}(s)$ can be rewritten as

$$\begin{gathered} \mathbf{d}(s) - \mathbf{d_0}(s) = \\ p_x\mathbf{U}(\mathbf{s}) \begin{bmatrix} \mathbf{1} \\ \mathbf{0} \end{bmatrix} + p_y\mathbf{U}(\mathbf{s}) \begin{bmatrix} \mathbf{0} \\ \mathbf{1} \end{bmatrix} + (L_{11} - 1)\mathbf{U}(\mathbf{s}) \begin{bmatrix} \mathbf{Q^x_0} \\ \mathbf{0} \end{bmatrix} + \\ + L_{12}\mathbf{U}(\mathbf{s}) \begin{bmatrix} \mathbf{Q^y_0} \\ \mathbf{0} \end{bmatrix} + L_{21}\mathbf{U}(\mathbf{s}) \begin{bmatrix} \mathbf{0} \\ \mathbf{Q^x_0} \end{bmatrix} + (L_{22} - 1)\mathbf{U}(\mathbf{s}) \begin{bmatrix} \mathbf{0} \\ \mathbf{Q^y_0} \end{bmatrix} \end{gathered}$$

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,

$$\mathbf{Q} - \mathbf{Q_0} = \mathbf{WX}$$

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

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*,

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

We use the active contour tracker of Blake *et al.* [3], 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 [6], 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,

$$\mathbf{R} = \begin{bmatrix} cos\psi & 0 & -sin\psi \\ 0 & 1 & 0 \\ sin\psi & 0 & cos\psi \end{bmatrix} \tag{10}$$

and

$$\mathbf{L} = \frac{Z_0}{Z_0 cos\psi + T_z} \begin{bmatrix} cos\psi & 0 \\ 0 & 1 \end{bmatrix} \tag{11}$$

$$\mathbf{p} = \frac{1}{Z_0 cos\psi + T_z} \begin{bmatrix} \alpha_u & 0 \\ 0 & \alpha_v \end{bmatrix} \begin{bmatrix} T_x - Z_0 sin\psi \\ T_y \end{bmatrix} \tag{12}$$

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

$$\mathbf{X} = \left(\frac{\alpha_u (T_x - Z_0 sin\psi)}{Z_0 cos\psi + T_z}, \frac{\alpha_v T_y}{Z_0 cos\psi + T_z}, \frac{Z_0}{Z_0 cos\psi + T_z} cos\psi - 1, \frac{Z_0}{Z_0 cos\psi + T_z} - 1, 0, 0 \right) \tag{13}$$

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,

$$cos\psi = \frac{L_{11}}{L_{22}} \tag{14}$$

$$\alpha_u \frac{T_x}{Z_0} = \frac{p_x}{L_{22}} + \alpha_u sin\psi \tag{15}$$

$$\alpha_v \frac{T_y}{Z_0} = \frac{p_y}{L_{22}} \tag{16}$$

$$\frac{T_z}{Z_0} = \frac{1}{L_{22}} - cos\psi \tag{17}$$

4.1 Discussion

These results keep the ambiguities usual in monocular images. Equations (15), (16), (17) 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. Equation (14) keeps the Necker reversal ambiguity. From $cos\theta$ only the magnitude of θ can be computed. The sign of the angle cannot be recovered.

The bas-relief ambiguity is cancelled as we assume that 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, and the ambiguity disappears. However, another ambiguity appears , namely the rotation– translation ambiguity, which is common when trying to compute 3D motion taking the reference coordinate system on the camera ?????????. The ambiguity arises because rotation about the Y axis and translation along the X axis produce similar effects as reflected in equation (15). Translation along the X axis is added to $\alpha_u sin\psi$, and the two terms can not be split unless one of them is known.

This ambiguity is the responsible for the invariance of $\frac{L_{11}}{L_{22}}$ to small 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 explain this ambiguity in [17], they prove that the images of points rotating around the Y axis of the camera describe hiperbolas 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 non local processing is applied (as proposed in [39]), or there are motion parallax effects in the observed regions [25]. A comparison between these methods is presented in [37].

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. Thus, the data provided by the compass (namely, the ψ angle) is combined with the analysis of the contour to provide the 3D egomotion estimation.

If the camera has not any degree of freedom, it is necessary to know $\alpha_u = fK_u$, in order to recover the scaled-translation. K_u can be assumed constant along the sequence, but f can change with focus or zoom. It seems too demanding to assume that f can be known. In order to avoid this assumption, the solution is to provide the camera with one degree of freedom, as we have stated in the introduction. The camera has to be 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

$$\mathbf{X} = \left(\frac{\alpha_u T_x}{Z_0 + T_z}, \frac{\alpha_v T_y}{Z_0 + T_z}, \frac{Z_0}{Z_0 + T_z} - 1, \frac{Z_0}{Z_0 + T_z} - 1, 0, 0\right) \tag{18}$$

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

$$\alpha_u \frac{T_x}{Z_0} = \frac{p_x}{L_{22}} \tag{19}$$

$$\alpha_v \frac{T_y}{Z_0} = \frac{p_y}{L_{22}} \tag{20}$$

$$\frac{T_z}{Z_0} = \frac{1}{L_{22}} - 1 \tag{21}$$

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 [33] or road vehicles [13], assuming motion on a planar surface.

We estimate the likely time to contact to 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 neither the size and distance of the target, nor the speed of the camera towards it.

From equation (21) we can observe that the scaled depth of the target can be computed as

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

Let us call this magnitude H_i for the contour at frame i

$$H_i = \frac{Z_0 + T_{zi}}{Z_0} \tag{23}$$

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

The difference between H in consecutive frames is

$$H_i - H_{i-1} = \frac{T_{zi} - T_{zi-1}}{Z_0} \tag{24}$$

Therefore,

$$(H_i - H_{i-1})\frac{1}{H_i} = \frac{T_{zi} - T_{zi-1}}{Z_0 + T_{zi}} = \frac{-1}{\tau} \tag{25}$$

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,

$$\tau = \frac{H_i}{H_{i-1} - H_i} \tag{26}$$

The implementation of the theory shows that this measure is a useful tool to predict the collision time. Figure 3 shows the initial image in a sequence taken while the viewer moves towards a target. In this case the target is the black square. This experiment was carried out inside a laboratory in order to estimate the reliability of the results. The sequence was recorded at a constant velocity of approximately 16cm per time unit, and the target was set at 97cm from the initial position. Figure 4 shows four samples of the sequence. The shape vectors for these examples are

Fig. 2. **Experiment to evaluate the TTC computation.**

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

$$\begin{aligned} X_A &= [0, 0, 0.22, 0.22, 0, 0] \\ X_B &= [0, 0, 0.56, 0.56, 0, 0] \\ X_C &= [0, 0, 1.19, 1.19, 0, 0] \\ X_D &= [0, 0, 2.69, 2.69, 0, 0] \end{aligned} \tag{27}$$

Figure 2 depicts the situation in which the experiment is set. Figure 5 plots the recovered TTC as a function of time. It can be observed that the graphic decreases linearly as predicted for a uniform motion. We detect a likely collision with the target when the heading direction points towards the target and the TTC is under a certain threshold.

6 Self-Calibration of the principal point

In this section we explain how the principal point can be computed from the analysis of contour deformation combined with a set of point matches.

The relation between the projections of a 3D point on different image planes

A B

C D

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 (approximately 16cm per time unit). 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.*

Fig. 5. **Estimated time to contact as a function of time.** *It can be observed that the plot decreases linearly as predicted for a uniform motion.*

fulfils the following equation in homogeneous notation (see [15] for details)

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

where Z_i is the depth of the 3D point. In particular, when the rotation has been compensated and the image has only the effects of the translation, the above equation simplifies to,

$$\mathbf{u^{(2)}} = \mathbf{u^{(1)}} + \frac{\mathbf{AT}}{\mathbf{Z_i}} = \mathbf{u^{(1)}} + \begin{bmatrix} \alpha_u \frac{T_x}{T_z} + u_0 \\ \alpha_v \frac{T_y}{T_z} + v_0 \\ 1 \end{bmatrix} \frac{T_z}{Z_i} \tag{28}$$

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 conventional notation, we have two linear equations with three unknowns (the components of the epipole and the relative depth $\frac{Z_i}{T_z}$)

$$\begin{aligned} u_x^{(2)} &= e_x - \frac{Z_i}{T_z}(u_x^{(2)} - u_x^{(1)}) \\ u_y^{(2)} &= e_y - \frac{Z_i}{T_z}(u_y^{(2)} - u_y^{(1)}) \end{aligned} \tag{29}$$

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 equation (30),

$$
\begin{aligned}
u_0 &= e_x - \alpha_u \frac{T_x}{T_z} \\
v_0 &= e_y - \alpha_v \frac{T_y}{T_z}
\end{aligned}
$$

This computation can be repeated each time we want the principal point updated. Initially, it can be computed off-line. In Section 8 we explain that, once the epipole is known, the analysis of the contour provides enough information to guide the matching between frames, and from the guided matches the principal point can be updated.

7 Computation of the heading direction. Finding the epipole.

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 on the image plane

$$
\begin{bmatrix} e_x \\ e_y \end{bmatrix} = \begin{bmatrix} \alpha_u \frac{T_x}{T_z} + u_0 \\ \alpha_v \frac{T_y}{T_z} + v_0 \end{bmatrix} \tag{30}
$$

From equations (19), (20), (21) we have,

$$
\alpha_u \frac{T_x}{T_z} = \frac{p_x}{1 - M_{22}} \tag{31}
$$

$$
\alpha_v \frac{T_y}{T_z} = \frac{p_y}{1 - M_{22}} \tag{32}
$$

that lead us to the heading direction using the principal point computed in the preceding section.

8 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 reference object. 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 different points in the image. 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 the depth variation of its points is small compared to their distances to the camera. However, this simplified model do not generally fit the rest of the image, particularly when the scene has objects at different depths.

A point $\mathbf{u}^{(\mathbf{1})}$ in the first image corresponds to a 3D point that lies on the ray that backprojects through $\mathbf{u}^{(\mathbf{1})}$. Therefore, its corresponding point in the second image, $\mathbf{u}^{(\mathbf{2})}$, should lie on the projection of this ray, namely, the epipolar line of $\mathbf{u}^{(\mathbf{1})}$. 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 [27,40,43]. It relates the projections $\mathbf{u}^{(\mathbf{1})}, \mathbf{u}^{(\mathbf{2})}$ of a 3D point, in homogeneous notation, as follows,

$$\mathbf{u}^{(\mathbf{2})\mathbf{T}}\mathbf{F}\mathbf{u}^{(\mathbf{1})} = 0. \tag{33}$$

The $\mathbf{F}$ can be split up [27,12,43] as

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

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 $\mathbf{A}^{-\mathbf{1}}$ 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, the epipoles are the same for both images and they equal the heading direction. In this case, $\mathbf{F}$ simplifies to

$$\mathbf{F} = \mathbf{A}^{-\mathbf{T}}[\mathbf{T}]_*\mathbf{A}^{-\mathbf{1}} = [\mathbf{AT}]_* = [\mathbf{e}^{(\mathbf{2})}]_* = [\mathbf{e}^{(\mathbf{1})}]_* \tag{34}$$

Using homogeneous notation, a line $\mathbf{l}^{(\mathbf{2})}$ passing through a point $\mathbf{u}^{(\mathbf{2})}$ fulfils the following equation (an introduction to perspective geometry can be found in [15])

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

Therefore, from equation (33), the epipolar line can be computed as

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

And, from equation (34),

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

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

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

Thus, it is proved that the epipolar line in the second image for a point in the first image is the line joining the epipole with the point in the first image.

The epipolar lines have been computed to be used as a guide for matching features between frames. Some results are shown in figures 6 to 9. figure If the disparity between images increases, an algorithm based 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.

9 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

Fig. 6. **Initial image.** *The square in the middle of the pattern were taken as target. The deformation of the contour fitted on it allows to recover the heading direction and epipolar lines.*

Fig. 7. **Template used in the experiment to recover the heading direction.** *AFEGIR FIG AMB EL CONTORN AJUSTAT !!!!*

Fig. 8. **Epipolar lines.** *Epipolar lines relating the image in figure 6 with the image recorded after a translation in the Z direction. The heading direction is the point of intersection of all epipolar lines.*

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

Fig. 9. **Epipolar lines.** *Epipolar lines relating image in figure 6 with the image recorded after a larger translation in the Z direction. The heading direction is the point toward the epipole, and the epipole is the point of intersection of all epipolar lines. The results are right even when there is a reduced region of the image in common with the original one.*

scene assuming that the surface is smooth.

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 equations (21) and (29), we can solve for the scaled depth,

$$
\begin{aligned}
\frac{Z_i}{Z_0} &= \frac{\left(e_x - u_x^{(2)}\right)\frac{T_z}{Z_0}}{u_x^{(2)} - u_x^{(1)}} = \frac{\left(e_x - u_x^{(2)}\right)\frac{1}{L_{22}}}{u_x^{(2)} - u_x^{(1)}} \\
\frac{Z_i}{Z_0} &= \frac{\left(e_y - u_y^{(2)}\right)\frac{T_z}{Z_0}}{u_y^{(2)} - u_y^{(1)}} = \frac{\left(e_y - u_y^{(2)}\right)\frac{1}{L_{22}}}{u_y^{(2)} - u_y^{(1)}}
\end{aligned}
$$

The epipole and $\frac{T_z}{Z_0}$ are the same for all points in the same frame. The magnitude of the depth of a point is,

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

where $\mathbf{e}$ is the epipole.

The above deduction is valid as far as there is a non nul translation in Z. The analysis of contour deformations allows to detect when the translation in Z is nul and use a different set of equations to solve for 3D scene structure, in this case. When there is no translation in Z equation (29) is reduced to,

Fig. 10. **Proposed scheme to recover epipolar geometry and 3D structure.**

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

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

Either if there is a translation in Z or not, the 3D reconstruction is improved by adding the points inside the target to the set of points for which the depth is known. From equation (21) we have an approximation of the depth of points inside the target

$$\frac{Z_0 + T_z}{Z_0} = 1 + \frac{1}{L_{22}} \tag{35}$$

Fig. 10 depicts the proposed scheme. We emphasise the fact that there is an computed. After this, a very simple scheme allows 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.

9.1 Experimental results

The proposed algorithm has been tested on several image sequences, and good results have been obtained. The results were evaluated at a first stage using indoor scenes, but they have been proved also with real outdoor scenes. In this paper we provide the qualitative depth map of one of these scenes. Fig. 11 and Fig. 12 show different frames of the scene. An active contour has been fitted to the target, which is drawn in red.

Fig. 13 highlights some points of the image, for which the epipolar lines are drawn in Fig. 14. A set of salient features are automatically detected in Fig. 15. They are matched with the ones in Fig. 12 using the epipolar lines computed from the analysis of the active contour. Once the matches are known, the 3D reconstruction is computed. Fig. 16 shows the 3D reconstruction. 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. 17 depicts a view of the final result.

Fig. 11. **Target.** *An active contour (in red) is fitted to the target.*

Fig. 12. **Target tracking.** *The target is tracked along the sequence.*

10 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 combining the matching of features between frames with the information derived from an active contour. The proposed method is based on a direct measure of image deformation from an active contour fitted to a reference object. It is essentially different from the common techniques that use velocity or displacement fields as the unique basis for further computation [22,37,34,41].

Fig. 13. **Image points.** *The epipolar lines will be computed for the set of salient points in blue.*

Fig. 14. **Epipolar lines** *The epipolar lines (in blue) corresponding to the image points in figure 13 are drawn.*

Fig. 15. **Whole set of image points.** *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.*

Several advantages are achieved from focussing the processing on the target. The first one is speed, the epipolar geometry is recovered at frame rate from live video using a Silicon Graphics Indy. The second one is the robustness of the method to independent motions in the scene. It is remarkable to observe that most of the current methods rely on the assumption of a single indepen-

Fig. 16. **3D structure recovery from interpolation of depth of matched points.**

Fig. 17. **3D structure recovery adding the depth of the points inside the target.** *This reconstruction has been computed interpolating the depth of the matched points and that of the points inside the target.*

dent motion; i.e., they work for scenes containing only one moving object or, alternatively, a moving camera in an stationary environment [23,43]. The third one is that the attentional mechanism allows to assume a simplified camera model for the region used as reference, no matter if this model does not fit the rest of the image.

The proposed method is based 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 these cases, the methods based only on point matches are complemented with the information of the contour and the results improve considerably. Traditional techniques require a initial set of reliable matches to extract the epipolar geometry and then guide the search for additional matches [27,40,43], while the proposed method takes profit out of the analysis of the contour to avoid the initial unguided search. On the other hand, a number of the previous works rely on the computation of the fundamental matrix, which become 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 [15,27]. The key question is to know when to switch from using the fundamental matrix to using a homography, and viceversa. 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 to that, the analysis of the active contour allows 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.

Further work has been planed to extend the method to use 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.

Acknowledgments

The authors wish to thank Andrew Blake for very useful discussions and guidance in the first stages of this work. This 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 Direccio General de Recerca of the Generalitat of Catalunya.

References

[1] P.A. Beardsley, A. Zisserman, and D.W. Murray. Sequential updating of projective and affine structure from motion. *Int. J. Computer Vision*, 1994.

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

[3] A. Blake, M.A. Isard, and D. Reynard. Learning to track the visual motion of contours. *J. Artificial Intelligence*, 78:101–134, 1995.

[4] T.J. Broida, S. Chandrashekhar, and R. Chellappa. Recursive estimation of 3-d kinematics and structure from a noisy monocular image sequence. In

Proceedings of ISCV, volume A New Approach to Image Feature Detection with Applications, November 1995.

[5] E. Celaya and F. Porta. Control of a six-legged robot walking on abrupt terrain. In *Proc. of IEEE International Conference on Robotics and Automation*, pages 2731–2736, Minneapolis, Minnesota, 1996.

[6] 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.

[7] 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.

[8] R. Cipolla, Y. Okamoto, and Y. Kuno. Robust structure from motion using motion parallax. In *4th Int. Conf. on Computer Vision*, 1993.

[9] J. Climent and A. Benito. Disseny de mascares patro adaptatives, per a la determinacio de l'orientacio d'un objecte mobil en sequencies d'imatges. In *1r Seminari de trevall en Automatica, Robotica i Percepcio*, Barcelona. Spain, 1996.

[10] J.L. Crowley, P. Bobet, and C. Schmidt. Maintaining stereo calibration by tracking image points. In *Proc. Conf. Computer Vision and Pattern Recognition*, 1993.

[11] A. Davison and D. W. Murray. Mobile robot localisation using active vision. In *Proc. 6th European Conf. Computer Vision*, 1998. Submitted.

[12] R. Deriche, Z. Zhang, Q.T. Luong, and O. Faugeras. Robust recovery of the epipolar geometry for an uncalibrated stereo rig. In *Proc. 3rd European Conf. Computer Vision*, pages 567–576, Stockholm, 1994.

[13] E.D. Dickmanns and V. Graefe. Dynamic monocular machine vision. *Machine Vision and Applications*, 1:223–240, 1988.

[14] E.D. Dickmanns, B. Mysliwetz, and T. Christians. An integrated spatio-temporal approach to automatic visual guidance of autonomous vehicles. *IEEE Trans. on Systems, Man and Cybernetics*, 20:6:1273–1284, 1990.

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

[16] O. Faugeras, F. Lustman, and G. Toscani. Motion and structure from motion from point and line matches. In *Proc. 1st Int. Conf. on Computer Vision*, pages 25–34, 1987.

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

[18] J. Fernandez and A. Casals. Aportacio a la deteccio de camins navegables en entorns naturals a partir de l'analisi de regions en sequencies d'imatges. In *1r Seminari de trevall en Automatica, Robotica i Percepcio*, Barcelona. Spain, 1996.

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

[20] E. Gat, F. Desai, R. Iulev, J. Loch, and D.P. Milter. Behaviour control for robotic exploration of planetary surfaces. *IEEE Trans on Robotics and Automation*, 10:4:490–503, 1994.

[21] J. Gonzalez and A.J. Munoz. Sistema de vision monocular para la estimacion de la posicion de un robot movil en entornos estructurados. In *Proc. of XVI Jornadas de Automatica. IV Reunion del Grupo de Vision Artificial*, pages 26–39, San Sebastian, Spain, 1995.

[22] C. Harris. Determination of egomotion from matched points. In *Proc. of 3rd Alvey Vision Conference*, 1987.

[23] M. Irani, B. Rousso, and S. Peleg. Recovery of egomotion using image stabilization. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, pages 454–460, Seatle, WA, June 1994.

[24] A. Kosaka and J. Pan. Purdue experiments in model based vision for hallway navigation. In *Proc of Workshop on Vision for Robots in IROS'95 Conference*, pages 87–96, 1995.

[25] J. Lawn93 and R. Cipolla. Epipole estimation using affine motion parallax. In *Proc. British Machine Vision Conf.*, 1993.

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

[27] Q.T. Luong, R. Deriche, O. Faugeras, and T. Papadopoulo. On determining the fundamental matrix: analysis of different methods and experimental results. Technical report, INRIA. Sophia Antipolis, 1993. RR 1894.

[28] 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, San Jose, California*, January 2000.

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

[30] J.L. Munday and A. Zisserman. *Geometric invariance in computer vision.* MIT Press, 1992.

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

[32] D. Pack. Perceptual-based control for a quadruped walking robot. In *Proc. of the IEEE International Conference on Robotics and Automation*, pages 2994–3001, 1996.

[33] J. Santos-Victor and G. Sandini. Visual behaviors for docking. Technical report, LIRA-Lab-DIST University of Genova, 1994. TR 2/94.

[34] C. Silva and J. Santos-Victor. Robust egomotion estimation from the normal flow using search subspaces. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 19:9, 1997.

[35] S. Soatto and P. Perona. Recursive estimation of camera motion from uncalibrated image sequences. In *Proc. 1st IEEE International Conference on Image Processing (ICIP)*, 1994.

[36] C. Taylor and D. Kriegman. Vision-based motion planning and exploration algorithms for mobile robots. In Goldberg, Halperin, Latombe, and Wilson., editors, *The algorithmic foundations of robotics.* A.K. Peters, 1995.

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

[38] E. Todt and C. Torras. Detection of natural landmarks through multiscale opponent features. In *Submitted to publication*, 1999.

[39] C. Tomasi and Shi J. Direction of heading from image deformations. In *IEEE Conference on Computer Vision and Pattern Recognition*, 1993.

[40] P. H. S. Torr and D. W. Murray. A review of robust methods to estimate the fundamental matrix. In *Int. J. Computer Vision*, 1996.

[41] A. Verri and E. Trucco. Finding the epipole from uncalibrated optical flow. In *Proc. British Machine Vision Conf.*, 1997.

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

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

[44] Z. Zhang. A new multistage approach to motion and structure estimation: From essential parameters to euclidean motion via fundamental matrix. Technical report, INRIA, Sophia Antipolis, France, 1996. No. 2910.