

Depth map from the combination of matched points with active contours

Elisa Martínez

Dep. de Comunicacions i Teoria del Senyal
Ramon Llull University
Pge. Bonanova, 8,
08022 Barcelona, Spain
elisa@salleurl.edu

Carme Torras

Institut de Robòtica
i Informàtica Industrial
Gran Capità, 2-4,
08034 Barcelona, Spain
ctorras@iri.upc.es

Abstract

This paper describes the analysis of an active contour fitted to a target in a sequence of images recorded by a freely moving uncalibrated camera. The motivating application is the visual guidance of a robot towards a target. Contour deformations are analysed to extract the scaled depth of the target, and to explore the feasibility of 3D egomotion recovery. The scaled depth is used to compute the time to contact, which provides a measure of distance to the target, and also to improve the common depth maps obtained from point matches, which are a valuable input for the robot to avoid obstacles.

1 Introduction

Active contours have proved to be a valuable tool for tracking a moving target observed by a static uncalibrated camera [2]. The analysis of an active contour has also been proposed to compute the 3D pose of a tracked target when the calibration parameters of the camera are known. Here we extend the analysis of an active contour to the case in which it is fitted to a static target, observed by an uncalibrated camera freely moving in 3D space.

First, we explore the feasibility of 3D egomotion recovery from the analysis of an active contour in a sequence recorded by an uncalibrated camera. Second, the depth of the target is recovered up to a scale factor, which suffices to compute a measure of distance to the target, namely the time to contact with the target. The usual approach to compute the time to contact is based on optic flow [9], either by obtaining the velocity vectors at all image positions, or by extracting some salient points and tracking them from frame to frame. Less works address the problem of estimating the time to contact on the basis of only local information. Cipolla and Blake [5] compute the surface orientation and time to contact using the area moments of closed contours extracted from the B-spline snake control points.

Their procedure can be used for qualitative visual navigation, if the viewer can make deliberate movements or has stereoscopic vision [6]. Here we extract the time to contact at frame rate directly from the shape vector of an active contour using a single freely moving camera. Finally, we prove that common depth maps obtained from point matches can be improved by adding the scaled depth of the points inside the target, which can be approximated by the recovered depth of the contour under weak- -perspective conditions.

The present work is part of a project aimed at the visual guidance of a low-cost robot walking towards a target [4] in an unstructured outdoor scene. It is a six-legged robot equipped with a single camera rigidly mounted to its body. Since our robot has deliberately limited resources, we do not look for very sophisticated procedures but instead we want to reach the best possible compromise between simplicity and performance.

The paper is structured as follows. The next section characterizes the projection of a contour using an uncalibrated weak-perspective camera model. Section 3 presents the derivation of the shape vector, from which we extract the scaled depth of the target (Section 4) and the time to contact with the target (Section 5). In Section 6 we combine the depth computed for matched point with the scaled depth of the target to improve the final depth map. Finally, the advantages and limitations of the proposed procedure are discussed in Section 7.

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 weak perspective conditions, i.e. when the object fits in a small field of view and the range 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

$$\mathbf{D}(s) = \mathbf{R}\mathbf{D}_0(s) + \mathbf{T}. \quad (1)$$

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

Taking the camera coordinate frame as reference, the projected contour on the image plane has the following expression

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

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, but we explicitly write them in order to highlight their effect.

Under weak-perspective conditions $Z_0(s)$ can be approximated by the average depth Z_0 of the contour, and $R_{31}X_0(s) + R_{32}Y_0(s) \ll R_{33}Z_0 + T_z$, then equation (2) can be rewritten as

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

In particular, the projection of the template is

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

Combining equations (3) and (4),

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

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

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

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

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

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

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

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

3 Affine deformation from the analysis of active contours

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

The contour is represented as a parametric spline, $\mathbf{d}(s) = (d_x(s), d_y(s))^T$, where both $d_x(s)$ and $d_y(s)$ are B-spline curves. We can write them as a function of their control points,

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

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

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

where $\mathbf{0}^T$ is a column vector of zeros, $\mathbf{U}(s) = \mathbf{I} \otimes \mathbf{B}(s)$ ¹ and \mathbf{Q} is the vector of control points. In particular, the template is written as

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

where \mathbf{Q}_0 is the vector of control points of the template. Then

$$\mathbf{d}(s) - \begin{bmatrix} u_0 \\ v_0 \end{bmatrix} = \mathbf{U}(s)\tilde{\mathbf{Q}}_0,$$

where $\tilde{\mathbf{Q}}_0$ is the vector of control points of the observed template minus the coordinates of its centre. Substituting this expression in equation (5), we obtain

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

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

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

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,

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

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 an active contour tracker based on the Kalman filter [2] 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.

¹ \otimes is the kronecker product.

4 Computation of scaled depth of the target

The shape vector computed in the preceding section has a direct relation with the egomotion parameters (equations (6) and (7)). From equation (6), we observe that the aspect ratio $\frac{K_u}{K_v}$ needs to be known in order to deduce the rotation parameters. It can be assumed to be known for off-the-shelf cameras. Without loss of generality, we assume that the aspect ratio is one. Then, the 3D motion parameters can be recovered writing the rotation matrix in terms of the Euler angles,

$$\mathbf{R} = \mathbf{R}_z(\phi)\mathbf{R}_x(\theta)\mathbf{R}_z(\psi) \quad (10)$$

where $\mathbf{R}_z(\psi)$ and $\mathbf{R}_x(\theta)$ are rotation matrices about the Z axis and $\mathbf{R}_x(\theta)$ is a rotation matrix about the X axis.

Substituting equation (10) into equation (6)

$$\mathbf{L} = \frac{Z_0}{R_{33}T_z + Z_0} \mathbf{R}_z|_2(\phi) \begin{bmatrix} 1 & 0 \\ 0 & \cos\theta \end{bmatrix} \mathbf{R}_z|_2(\psi) \quad (11)$$

and,

$$\mathbf{L}\mathbf{L}^T = \mathbf{R}_z|_2(\phi) \begin{bmatrix} \left(\frac{Z_0}{R_{33}T_z + Z_0}\right)^2 & 0 \\ 0 & \left(\frac{Z_0 \cos\theta}{R_{33}T_z + Z_0}\right)^2 \end{bmatrix} \mathbf{R}_z|_2^{-1}(\phi). \quad (12)$$

Then

$$\cos\theta = \sqrt{\frac{\lambda_2}{\lambda_1}}, \quad (13)$$

where λ_1 is the largest eigenvalue of $\mathbf{L}\mathbf{L}^T$ and λ_2 the smallest one. The scaled depth of the target is

$$\frac{R_{33}Z_0 + T_z}{Z_0} = \frac{1}{\sqrt{\lambda_1}}. \quad (14)$$

The angle ϕ can be extracted from the eigenvectors of $\mathbf{L}\mathbf{L}^T$. The eigenvector \mathbf{v}_1 with largest eigenvalue equals the first column of $\mathbf{R}_z|_2(\phi)$,

$$\mathbf{v}_1 = \begin{bmatrix} \cos\phi \\ \sin\phi \end{bmatrix}.$$

Isolating $\mathbf{R}_z|_2(\psi)$ from equation (11),

$$\mathbf{R}_z|_2(\psi) = (R_{33} + \frac{T_z}{Z_0}) \begin{bmatrix} 1 & 0 \\ 0 & \frac{1}{\cos\theta} \end{bmatrix} \mathbf{R}_z|_2(-\phi)\mathbf{L},$$

and observing, from equation (14), that

$$R_{33} + \frac{T_z}{Z_0} = \frac{1}{\sqrt{\lambda_1}},$$

we can find $\sin\psi$ and then ψ .

Once the angles ψ, θ, ϕ are known, the rotation matrix \mathbf{R} can be computed as in equation (10). The scaled translation along Z can be computed as

$$\frac{T_z}{Z_0} = \frac{1}{\sqrt{\lambda_1}} - R_{33}. \quad (15)$$

The rest of the components of the 3D translation vector cannot be computed without knowledge of the internal calibration parameters. Note that from equation (7),

$$\frac{T_x}{Z_0} = \frac{p_x}{\alpha_u \sqrt{\lambda_1}} - R_{13}, \quad (16)$$

$$\frac{T_y}{Z_0} = \frac{p_y}{\alpha_v \sqrt{\lambda_1}} - R_{23}. \quad (17)$$

The results keep the ambiguities common in monocular sequences. The Necker reversal ambiguity can be observed in equation (13) and equation (15) shows the depth-scale ambiguity. The former is reduced using the continuity of the sequence, but the latter can not be avoided unless the depth of the target is known or a stereo vision system is used. Nevertheless the results suffice to compute the time to contact with the target and the scaled depth map, as explained in the following sections.

5 Computation of the time to contact

The time to contact (TTC) with the target 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 [9] or road vehicles [7], assuming motion on a planar surface. Even when the motion is non-uniform, the profile of the time to contact as a function of time provides an important cue for braking reactions [3].

We estimate the likely time to contact to the target from the shape vector. 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 (14) and the recovered rotation we define H_i for the contour at frame i as

$$H_i \triangleq \frac{Z_0 + T_{zi}}{Z_0}$$

where T_{zi} is the translation in Z at frame i . The difference between H in consecutive frames is

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

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 evaluate the reliability of the results. A sequence was recorded at a constant velocity of approximately 16cm per time unit, and the target was set at 97cm 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 [2]. Fig. 1 shows different samples of the sequence. At each frame the active contour is fitted to the target. Figure 2 plots the recovered time to contact as a function of time. It can be observed that the graphic decreases linearly as predicted for a uniform motion. The experiment has been successfully repeated for more general motions that include translations parallel to the image plane and rotations. However, note that the computation of the TTC makes sense only for translational motions towards the target.

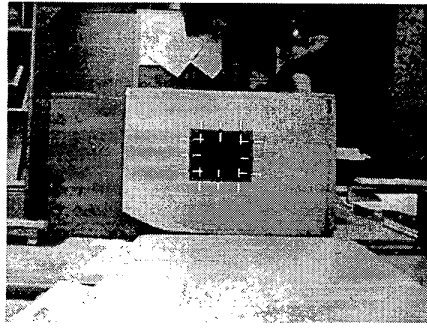
6 Depth map

There are several works that deal with the recovery of 3D structure from point matches [8, 1]. We propose to improve the depth map by adding the scaled depth of points inside the target to the set of scaled depths computed for matched points.

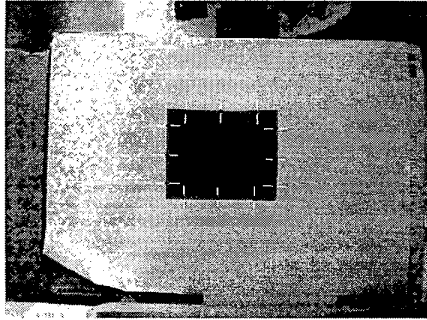
First, salient points and their matches between frames are detected [10] and their scaled depths ($\frac{Z_i}{T_i}$) are computed. Then, the 3D map is complemented by adding the scaled depth of points inside the target to the set of scaled depths computed for matched points. Under weak perspective conditions, the scaled depth of points inside the target can be approximated by the scaled depth of their contour (Equation (14)).

The proposed scheme has been tested, at a first stage, using indoor scenes, and later it has been successfully applied to real outdoor scenes. Here we provide the depth map of one of these scenes. Fig. 3 shows a sample of the sequence, in which the active contour is fitted to the target (i.e. the door).

A set of salient features for the frame in Fig. 3 are automatically detected [10] in Fig. 4. Once the matches for these points are found in the following image of the sequence, their scaled depth is computed. The final depth map is calculated by the interpolation of the depth of these points and the depth of the points inside the target. Figure 5 depicts a



A



B

Figure 1: Estimation of TTC from the deformation of an active contour. We observe the first (A) and the last (B) images 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. 2).

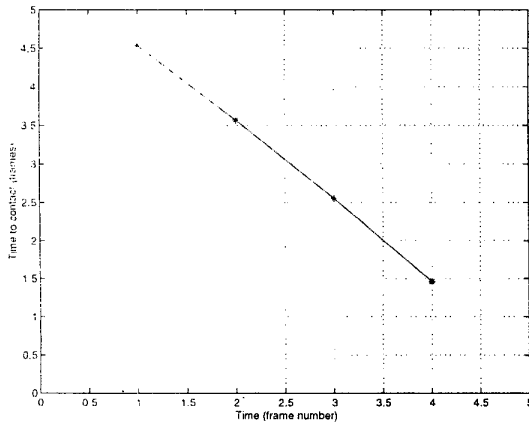


Figure 2: Estimated time to contact as a function of time. It can be observed that the plot decreases linearly as should happen for a uniform motion.

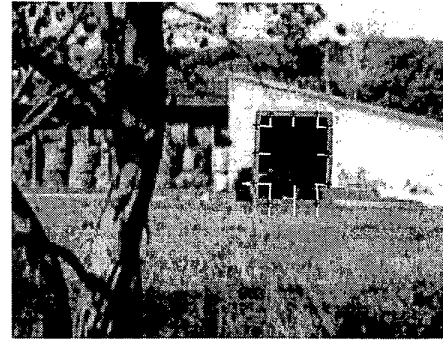


Figure 3: An active contour is fitted to the target and it is tracked along the sequence.

view of the final result. Blue colours depict distant regions while red ones represent nearby points. For instance, the silhouette of the tree can be observed, in red, in the left side of the image. The target can be observed in blue and 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. These data on the environment structure constitutes a valuable information for the viewer (e.g., a robot) to decide its best path towards the target.



Figure 4: Set of salient points on the image.

7 Concluding Remarks

This paper presents a new approach to provide depth information on a target. The proposed method is based on a direct measure of image deformation from an active contour fitted to the target. We deduce the shape vector of the active contour assuming a freely moving uncalibrated camera, and we relate this shape vector with the 3D egomotion of the camera. From this relation we discuss the recovery of egomotion and target position from the shape vector. The depth of the target is recovered up to a scale factor. The latter suffices to compute the time to contact

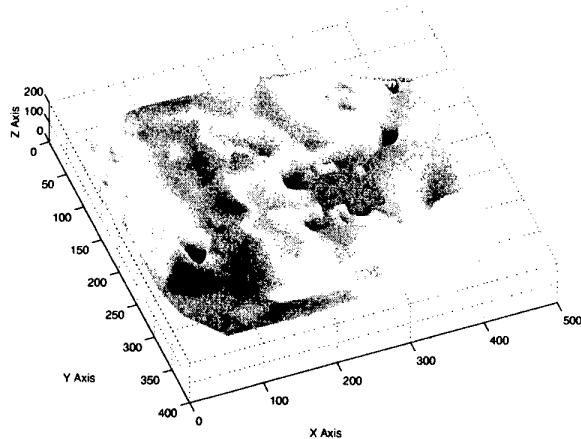


Figure 5: Final 3D depth map computed by interpolating the depth of the matched points and that of the points inside the target.

with the target, which is a measure of the depth of the target. In addition, the scaled depth of the target is used to improve the depth map of the environment obtained from matched points. As we assume weak-perspective conditions, we approximate the depth of points inside the target by the estimated depth of its contour, and the depth of all these points is added to the set of depths computed for the matched points. Therefore, the final depth map improves from that computed by using only the depth of matched points.

Several advantages are attained by focusing the initial processing on the target. The first one is speed, the scaled depth and time to contact are recovered at frame rate from live video using a Silicon Graphics Indy (150MHz). The second one is the robustness of the method to independent motions in the scene. The third one is that the focusing mechanism allows 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 only limitation is that the target has to be static and visible under a weak-perspective assumption, which is common, for instance, in applications in which a robot has to reach a distant target.

Further work has been planned to extend the method to use several contours automatically fitted to different regions in the image. The final depth map is expected to combine the depth of matched points with the depth of the contours. 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 homogeneously detected.

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 Direcció 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*, Volume 23, Number 3, pages 235–259, 1997.
- [2] A. Blake and M. Isard. *Active contours*. Springer, 1998.
- [3] V. Bruce, P.R. Green and M.A. Georgeson (editors). *Visual perception. Physiology, Psychology, and Ecology*. Psychology Press, 1996.
- [4] E. Celaya and F. Porta. A control structure for the locomotion of a legged robot on difficult terrain. *IEEE Robotics and Automation Magazine*, Volume 5, pages 43–51, 1998.
- [5] 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.
- [6] R. Cipolla and J. Hollinghurst. Visual robot guidance from uncalibrated stereo. In M. Brown and D. Terzopoulos (editors), *Real-time Computer Vision*, pages 169–187. Cambridge University Press, 1994.
- [7] E.D. Dickmanns and V. Graefe. Dynamic monocular machine vision. *Machine Vision and Applications*, Volume 1, pages 223–240, 1988.
- [8] 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*, pages 314–320, 1995.
- [9] J. Santos-Victor and Giulio Sandini. Divergent stereo in autonomous navigation: from bees to robots. In *Int. J. Computer Vision*, pages 159–177, 1995.
- [10] Z. Zhang, R. Deriche, O. Faugeras and Luong Q.T. A robust technique for matching two uncalibrated images through the recovery of the unknown epipolar geometry. *Artificial Intelligence Journal*, Volume 78, pages 87–119, 1995.