



Velocity-Guided Tracking of Deformable Contours in Three Dimensional Space

REUVEN ZARITSKY

Department of Electrical Engineering, Technion—Israel Institute of Technology, Haifa 32000, Israel

NATAN PETERFREUND

*Harmonic Inc., and Center for Engineering Science Advanced Research, Oak Ridge National Laboratory,
Oak Ridge, TN 37831-6355, USA*

v4p@ornl.gov

NAHUM SHIMKIN

Department of Electrical Engineering, Technion—Israel Institute of Technology, Haifa 32000, Israel

shimkin@ee.technion.ac.il

Received February 14, 2000; Revised January 10, 2002; Accepted September 20, 2002

Abstract. This paper presents a 3D active contour model for boundary detection and tracking of non-rigid objects, which applies stereo vision and motion analysis to the class of energy-minimizing deformable contour models, known as snakes. The proposed contour evolves in three-dimensional space in reaction to a 3D potential function, which is derived by projecting the contour onto the 2D stereo images. The potential function is augmented by a kinetic term, which is related to the velocity field along the contour. This term is used to guide the inter-image contour displacement. The incorporation of inter-frame velocity estimates in the tracking algorithm is especially important for contours which evolve in 3D space, where the added freedom of motion can easily result in loss of tracking. The proposed scheme incorporates local velocity information seamlessly in the snake model, with little computational overhead, and does not require exogenous computation of the optical flow or related quantities in each image. The resulting algorithm is shown to provide good tracking performance with only one iteration per frame, which provides a considerable advantage for real time operation.

Keywords: deformable contours, three-dimensional tracking, velocity snakes, optical flow, stereo vision

1. Introduction

Deformable contours have emerged in the last decade as a major tool for low level vision functions, including edge detection, segmentation, and tracking of non-rigid objects. Boundary detection and tracking based on deformable planar contours, known as snakes, were introduced in Kass et al. (1987). Energy-minimizing active contours are deformable contours that evolve under the influence of image-induced

potential, subject to certain internal deformation constraints. The contour dynamics may be specified by the Euler-Lagrange equations of motion associated with the contour potential. Using the image gradient as the potential function, for example, results in edge-seeking forces, leading the contour towards high contrast boundaries. Many variations and extensions of the original snake model have been reported in recent literature, see Blake and Yuille (1992), McInerney and Terzopoulos (1996), Blake and Isard

(1998), Jain et al. (1998), Sapiro (2001) and references therein.

In this paper we present an active contour model in three-dimensional space, which is used for temporal tracking of the contour of moving objects based on stereo image sequences.

Contour tracking in a sequence of images may be simply performed by initializing the snake at each new image according to its final position in the previous frame. A significant improvement can be obtained by incorporating a predictive step, which uses inter-frame velocity estimates to enhance the initial positioning. By positioning the snake closer to the actual object boundary, one obtains improved convergence and reduced likelihood of trapping in local minima. The required velocity estimates may be obtained by extrapolating the contour motion from previous images, using Kalman filtering or related methods; see e.g. Terzopoulos and Szeliski (1992) and Blake and Isard (1998). Another approach, more relevant to the present study, is to extract an estimate of local or global velocity from two consecutive images. In Basclé and Deriche (1993a), the authors use correlation and rigid motion models to estimate the global motion of the object between the two frames. Several papers (including Caselles and Coll 1996; Basclé et al., 1994; Berger et al., 1995) propose an explicit calculation of the local velocity field along the detected contour using optical flow methods. A related idea (Paragios and Deriche, 2000) is to use a potential term related to the temporal difference between images, in order to focus the tracking contour on regions of temporal change, as an indicative of object motion against a stationary background. In Bertalmio et al. (2000), two coupled PDEs deform one image into the other, while the derived velocity is used to transform the curve to its new position.

Recently, an integrated spatio-temporal snake model was proposed for 2D contour tracking in combined spatio-velocity space (Peterfreund, 1999). This *velocity snake* uses the optical flow constraints in order to propagate the contour in the direction of the local velocity field. Explicit computation of the optical flow is avoided by incorporating the flow constraint equation directly into the snake dynamics equations. The theoretical improvement in tracking performance of this model was demonstrated in Peterfreund (1997), which showed in a continuous time setting that zero tracking error is obtained for boundaries moving at a constant velocity. The present paper generalizes these ideas

to 3D contour tracking of shape and motion in stereo images.

Deformable contours are well suited for stereo matching, as they can perform simultaneously the two tasks of feature detection and correspondence. This was originally realized in Kass et al. (1987), which proposes to use different 2D contours in each of the stereo images, coupled by a stereo disparity smoothness term in their respective potential functionals. A deformable spline model for stereo matching that evolves in 3D was proposed in Basclé and Deriche (1993b), which employs an additive potential function based on the projections of the curve on each image. A similar stereoscopic potential was previously used in Terzopoulos et al. (1988) in the context of a deformable surface model. A similar idea will be employed in the tracking scheme proposed here. In Cham and Cipolla (1997), the authors propose an affine epipolar geometry scheme for coupling pairs of active contours in stereo images to enhance stereo tracking of 3D objects. Deriche et al. (1998) use stereo vision and geodesic active contours to trace 2D planar curves that lie at the intersection of the observed scene with a given plane in 3D space.

The related topic of deformable surfaces has been addressed, e.g., in Caselles et al. (1997) and Faugeras and Keriven (1999). While these models may be applied to three-dimensional object tracking, they may be too expensive computationally for real time operation.

The active contour model proposed here is a parameterized curve which evolves in three-dimensional space under the influence of a three-dimensional potential function. The 3D potential function is obtained by first projecting the 3D point onto the respective image planes, and then combining the two image potentials at the respective projected points. The 3D potential is essentially obtained by combining the two image potentials, evaluated at the projections of a 3D point onto the respective image planes. The basic potential we consider is the additive one, augmented with certain normalization factors that compensate for different sensitivities along the spatial axes. Furthermore, we examine in a theoretical framework certain properties and variations of the 3D potential function. To enhance tracking performance, the snake model is augmented by a velocity term related to the optical flow in each of the images. This term provides an additional force which approximately guides the snake contour along the actual contour motion. The resulting dynamic equation

of the tracking contour is realized in discrete time and space. In the discretized version, the velocity terms are employed in the first iteration of each new frame, possibly followed by additional potential-guided iterations. Even with a single iteration step per frame, the algorithm provides good tracking of realistic motion at video rate.

The underlying snake model in this paper is the classical energy-minimizing model, which evolves according to Newtonian dynamics. Recent work has advanced the so-called geodesic active contours, that utilize geometric curve evolution techniques, and enjoy important advantages (Kichenassami et al., 1996; Caselles et al., 1997; Sapiro, 2001). Contrary to these models, however, the classical snake preserves the curve parameterization (i.e., the identity of individual points) over subsequent iterations. This facilitates a natural incorporation of inter-frame velocities in the model. Furthermore, the linear evolution of the classical snake allows a relatively large step size, which is used to advantage in this work.

The proposed algorithm is easily scalable to more than two cameras, as well as to general geometric configurations, simply by considering the appropriate projections of the contour onto each image plane. We assume that the cameras are fully calibrated. Clearly, calibration errors will lead to errors in the spatial position estimates similar to the standard stereo problem.

The rest of the paper is organized as follows. In the next section we briefly review the velocity snake model in two dimensions. The three-dimensional model is presented in Section 3, followed by a brief investigation of the 3D potential and its properties in Section 4. Section 5 discusses some implementation considerations. The experimental results are presented in Section 6, followed by our concluding remarks.

2. Review of the 2D Velocity Snake

We start with a brief review of the two-dimensional velocity snake, introduced in Peterfreund (1999). Consider the collection $\{c(\cdot, t)\}$ of closed contours with coordinates $c(s, t) = (x(s, t), y(s, t))^T$, where $s \in [0, 1]$ is the position parameter and $t \in [0, \infty)$ a time parameter. As is usual we start out with continuous time and space parameters, the transition to an actual image sequence will be carried out through discretization and finite difference approximations. The snake Lagrangian, as presented in Terzopoulos and Szeliski

(1992), is given by:

$$L = \frac{1}{2} \int_0^1 \mu(s) |c_t|^2 ds - \frac{1}{2} \int_0^1 (w_1(s) |c_s|^2 + w_2(s) |c_{ss}|^2) ds - \frac{1}{2} \int_0^1 p(c, t) ds, \quad (1)$$

where c_x denotes the partial derivative with respect to the variable x , and the dependence of c on (s, t) is suppressed in our notation. The first term in the Lagrangian is the kinetic energy, where $\mu(s)$ is interpreted as the mass density of the snake at position s . The second term defines the internal deformation energy where $w_1(s)$ controls the tension and $w_2(s)$ controls the rigidity of the snake at s . The third term is the potential energy of the snake, associated with an image-induced potential function. The potential may be derived from a single (fixed) image, in which case the snake will converge to a static shape, or it may be derived from a temporal sequence of images for tracking purposes. In the latter case the potential becomes a function of time. In this paper, we shall consider for concreteness the common case of an edge seeking potential:

$$p(x, y, t) = -k \|\nabla (G_\sigma * I(x, y, t))\|, \quad (2)$$

which attracts the snake contour to the image edges. $I(x, y, t)$ denotes the brightness of pixel (x, y) in the image at time t , G_σ denotes a Gaussian smoothing filter with variance σ , ∇ is the position gradient (considered as a column vector), and $\|\cdot\|$ is the Euclidean norm. For additional potential terms which may be useful for tracking see, e.g., Paragios and Deriche (1999). Note that we use the *same* time variable for the snake evolution and the image sequence.

The Lagrangian (1) is complemented by an energy dissipation function which accounts for the effect of nonconservative forces. For the velocity snake this function is defined as

$$D = \frac{\gamma}{2} \int_0^1 \|L^T (c_t - v^b)\|^2 ds + \frac{\beta}{2} \int_0^1 \left\| \frac{\partial}{\partial s} c_t \right\|^2 ds \quad (3)$$

where L is a weight vector, and γ, β are positive constants. $v^b = v^b(s, t)$ is a “reference velocity” term, which ideally should equal the velocity of the tracked boundary; the actual choice will be specified below.

The related term can be interpreted as a viscous friction of the snake (moving at velocity c_t) against a background moving with velocity v^b . The second term represents a motion smoothness constraint. With the Lagrangian (1) and the dissipation function (3), the Euler-Lagrange equation of motion of the velocity snake is

$$\begin{aligned} \mu(s)c_{tt} + \gamma d(c_t, v^b) - \beta \frac{\partial}{\partial s} \left(\frac{\partial}{\partial s} c_t \right) \\ - \frac{\partial}{\partial s} (w_1 c_s) + \frac{\partial^2}{\partial s^2} (w_2 c_{ss}) = -\nabla p(c, t) \end{aligned} \quad (4)$$

where $d(c_t, v^b) = LL^T(c_t - v^b)$. Here and in the remainder of this paper we assume that the tension and rigidity parameters w_1 and w_2 are constant and do not depend on the position parameter s . The five terms on the left hand side stand, respectively, for the snake inertia; the exogenous velocity constraint; a temporal smoothness constraint; and spatial tension and rigidity terms. The potential gradient term provides attractive forces to desired features in the image, in our case to image edges.

Since the object's boundary velocity v^b is unknown, we use instead the apparent velocity (or *optical flow*) v^o of the image at the current contour position. This relies on the assumption that the contour is sufficiently close to the object's boundary. Moreover, an explicit calculation of the apparent motion field is avoided by an appropriate choice of the weight matrix L , which we define as $L \triangleq \nabla I(c(s, t), t)$. With this choice, the kinetic term in (4) becomes

$$\begin{aligned} d(c_t, v^b) \cong d(c_t, v^o) = \nabla I \nabla I^T (c_t - v^o) \\ = \nabla I (\nabla I^T c_t + I_t), \end{aligned} \quad (5)$$

where the last equality follows from the optical flow equation (Horn, 1986),

$$\nabla I(x, y, t)^T v^o + I_t(x, y, t) = 0. \quad (6)$$

Substituting (5) into (4) we obtain the velocity snake equation:

$$\begin{aligned} \mu(s)c_{tt} + \gamma \nabla I (\nabla I^T c_t + I_t) - \beta \frac{\partial}{\partial s} \left(\frac{\partial}{\partial s} c_t \right) \\ - \frac{\partial}{\partial s} (w_1 c_s) + \frac{\partial^2}{\partial s^2} (w_2 c_{ss}) = -\nabla p(c, t). \end{aligned} \quad (7)$$

Apparently, this equation does not require an explicit calculation of the velocity field v^o .

3. The 3D Velocity Snake Model

Given a sequence of stereo images, we extend here the two-dimensional velocity snake to object tracking in three spatial dimensions. This model requires a 3D potential function, which is extracted from the two stereo images by appropriate projections. The velocity term which serves to improve tracking performance across subsequent images will be defined based on the apparent motion in the image pair. We start with a brief definition of the relevant projection operations. The 3D tracking model will then be derived.

3.1. 3D Projections

We assume the general stereo camera configuration described in Fig. 1. Let $M = (X, Y, Z)^T$ denote a 3D space point in some fixed Cartesian coordinate frame, and let $m_\ell = (x_\ell, y_\ell)^T$ and $m_r = (x_r, y_r)^T$ denote the image coordinates in the "left" and "right" retinal image planes, respectively. We assume that the following transformations from world coordinates to image coordinates are available:

$$m_k = D_k(M), \quad k = \ell, r. \quad (8)$$

Further, let

$$\dot{m}_k = H_k(M)\dot{M} \quad (9)$$

denote the relation between a velocity vector \dot{M} and its projection \dot{m}_k on image k , with $H_k = \frac{\partial D_k}{\partial M}$.

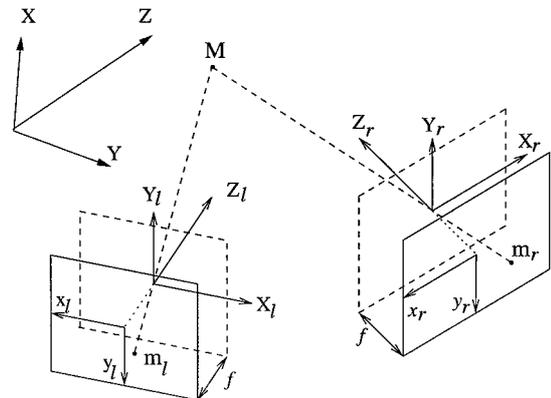


Figure 1. The stereo camera configuration. The focal planes (X_k, Y_k) are parallel to the retinal planes (x_k, y_k) . f is the focal length of each camera.

For concreteness, let us specify the standard equations for perspective projection. Let $M_k = (X_k, Y_k, Z_k)^T$, $k = \ell, r$, denote the world coordinate systems attached to the left and right cameras, respectively, each with its origin at the focal point and the (X, Y) plane parallel with the image plane (see Fig. 1). Let R_k be a rotation matrix and b_k a translation vectors such that

$$M_k = R_k M + b_k, \quad k = \ell, r. \quad (10)$$

Under perspective projection, the point M_k projects onto the image point $m_k = (x_k, y_k)^T$ according to

$$x_k = \frac{X_k}{Z_k} f, \quad y_k = \frac{Y_k}{Z_k} f, \quad (11)$$

where f denotes the focal length of the camera. (We use for simplicity identical focal lengths in the two cameras. This can always be obtained by a simple normalization of the image coordinates.) Combining these relations, we have

$$m_k = \frac{f}{Z_k} \begin{bmatrix} X_k \\ Y_k \end{bmatrix} \triangleq D_k(M), \quad k = \ell, r. \quad (12)$$

Differentiating Eq. (12) with respect to time, we obtain (9) with

$$H_k = \frac{\partial m_k}{\partial M_k} \frac{\partial M_k}{\partial M} = \begin{bmatrix} \frac{f}{Z_k} & 0 & -\frac{X_k}{Z_k^2} f \\ 0 & \frac{f}{Z_k} & -\frac{Y_k}{Z_k^2} f \end{bmatrix} R_k. \quad (13)$$

In our experiments we shall use the basic stereo configuration where two identical cameras share the same orientation, and are separated a distance $b > 0$ along the x axes. The coordinate system M is placed in the midpoint between the two camera coordinate systems. Referring to (10), we have in this case

$$\begin{aligned} R_\ell = R_r &= I \\ b_\ell &= \begin{bmatrix} +\frac{b}{2} \\ 0 \end{bmatrix} & b_r &= \begin{bmatrix} -\frac{b}{2} \\ 0 \end{bmatrix} \end{aligned} \quad (14)$$

where I denotes the identity matrix.

3.2. The Tracking Model

We define the snake at time t as a 3D parametric curve $C(s, t)$ in Cartesian coordinates, $C(s, t) =$

$[X(s, t), Y(s, t), Z(s, t)]$, where $s \in [0, 1]$. The corresponding Lagrangian is defined, similarly to the 2D case, as:

$$\begin{aligned} L &= \frac{1}{2} \int_0^1 \mu |C_t|^2 ds \\ &\quad - \frac{1}{2} \int_0^1 (w_1 |C_s|^2 + w_2 |C_{ss}|^2) ds \\ &\quad - \frac{1}{2} \int_0^1 P(C, t) ds. \end{aligned} \quad (15)$$

The first term of this equation is the 3D kinetic energy, the second term defines the internal deformation energy of the snake, where w_1 controls the tension and w_2 controls the rigidity. Note that we omit in this model the torsion term (which exists in three dimensions, but not in two) in order to simplify the resulting equations. Our experiments have indicated that this term does not contribute to tracking performance.

The third term in Eq. (15) is the potential energy of the contour, which needs to be computed based on the stereo image pair. As the contour C is three-dimensional, the potential field P needs to be defined in three dimensional space. This 3D potential energy is obtained by first projecting the contour unto each image, and then combining the pair of two-dimensional potentials that are obtained in an image plane to form a single potential function. Thus

$$P(M) = G(p_\ell(m_\ell), p_r(m_r)), \quad (16)$$

where $m_\ell = D_\ell(M)$ and $m_r = D_r(M)$ are the projections of the point M onto image coordinates, as defined in (8). The potentials p_ℓ and p_r are two-dimensional image potentials defined over the respective images, e.g. as in (2). G is an appropriate *potential fusion* function, that combines the two image potentials into a spatial potential.

The desired effect of this potential is to move the 3D contour so that its projections are attracted to relevant image features, such as image edges. The most obvious way to combine the two image potentials is a simple addition:

$$P(M) = p_\ell(m_\ell) + p_r(m_r). \quad (17)$$

Discussion of other alternatives and desirable properties of the potential function will be postponed to Section 4.

In analogy with the 2D case we propose the following 3D dissipation function:

$$\begin{aligned} D(C_t, V^b) &= \frac{\gamma}{2} \int_0^1 \|L_\ell^T(C_t - V^b)\|^2 ds \\ &\quad + \frac{\gamma}{2} \int_0^1 \|L_r^T(C_t - V^b)\|^2 ds \\ &\quad + \frac{\beta}{2} \int_0^1 \left\| \frac{\partial}{\partial s} C_t \right\|^2 ds \end{aligned} \quad (18)$$

where γ and β are positive scalars, $V^b = V^b(C(s, t))$ denotes an exogenous estimate of the 3D boundary velocity that corresponds to the snake point $C(s, t)$, and L_ℓ and L_r are real matrices which will later be defined as functions of the left and right image data. The third term represents a smoothness constraint. The first two terms give rise to a velocity-induced force in 3D space. In the following we show how these terms may be applied without an explicit computation of V^b .

Given the Lagrangian (15) and the dissipation function (18), the Euler-Lagrange equations of motion are:

$$\begin{aligned} \mu C_{tt} + \gamma(d_\ell(C_t, V^b) + d_r(C_t, V^b)) - \beta \frac{\partial}{\partial s} \left(\frac{\partial}{\partial s} C_t \right) \\ - \frac{\partial}{\partial s} (w_1 C_s) + \frac{\partial^2}{\partial s^2} (w_2 C_{ss}) = -\nabla P(C(s, t), t) \end{aligned} \quad (19)$$

where:

$$d_k(C_t, V^b) = L_k L_k^T (C_t - V^b), \quad k = \ell, r.$$

An explicit calculation of the 3D object velocity V^b is obviously a demanding task. This may be avoided by defining the following image dependent weight matrices: $L_\ell^T = \nabla I_\ell^T H_\ell$ and $L_r^T = \nabla I_r^T H_r$, where H_ℓ and H_r are the projection matrices in (9) and I_ℓ and I_r are the left and right image grey-scale intensities. This gives:

$$\begin{aligned} d_k(C_t, V^b) &= H_k^T \nabla I_k (\nabla I_k^T H_k C_t - \nabla I_k^T H_k V^b), \\ k &= \ell, r. \end{aligned} \quad (20)$$

The quantity $v_k^b = H_k V^b$ is the projected boundary velocity onto image k . We may approximate v_k^b by the apparent motion v_k^o of image k at the projected snake position. Using the optical flow constraint Eq. (6) for each image, the related term reduces to $\nabla I_k^T H_k V^b \cong$

$\nabla I_k^T v_k^o = -(I_k)_t$. Now (20) becomes:

$$d_k(C_t, V^b) = H_k^T \nabla I_k (\nabla I_k^T H_k C_t + (I_k)_t), \quad k = \ell, r. \quad (21)$$

The resulting Euler-Lagrange equation given by (19) and (21) is easily implemented by direct discretization, as discussed in Section 5.

4. Potential Shaping

We consider in this section the 3D potential function $P(M)$ that was defined in (16) as a combination of the two image potentials. Several properties and synthesis options will be elaborated. In the next subsection we consider the general potential of this form under the natural requirement of monotonicity of the combined potential in each of its components, and discuss the relation between minima of the image potentials and minima of the combined potential, both for points and for 3D curves. We then compare two specific potential functions, the additive and the multiplicative ones. Some practical considerations that raise the need for normalization along the Z axis for the additive potential are treated in Section 4.3. In the final part of this section we consider some mathematical criteria for selection of potential functions (or their gradients).

4.1. Preservation of Local Minima

We consider a potential function of the form (16), namely

$$P(M) = G(p_\ell(m_\ell), p_r(m_r)). \quad (22)$$

Recall that the potential field pulls the 3D snake in the direction of decreasing potential. Obviously, it is desirable that the induced motion of the projected image curves would be in the direction of decreasing image potentials. In particular, a natural requirement for the 3D potential is to inherit the minima of the image potentials: that is, the (local) minima of the 3D potential function should correspond to simultaneous minima of the two image potentials. We show that this is indeed the case provided only that the potential fusion function is increasing in each of its variables. We then discuss a similar minima-preservation property for 3D curves, and show that it holds provided the image gradient along each projected curve is small.

We assume henceforth that the two image potentials p_ℓ and p_r are smooth (i.e., continuously differentiable), and that the projection operators D_ℓ and D_r of (8) are uniformly continuous over the domain of interest. We further assume that the stereo configuration is not degenerate, in the sense that each pair (m_ℓ, m_r) of image points can correspond to at most one point M ; that is, the map (D_ℓ, D_r) is injective. We further assume that the potential fusion function G is smooth.

In the following, the term *minimum* refers to a strict local minimum: $f(x^*) > f(x)$ for all points $x \neq x^*$ in some neighborhood of x^* . A *weak* minimum refers to the weak local minimum ($f(x^*) \geq f(x)$ in some neighborhood).

Claim 1. Suppose that the potential fusion function $G(\cdot, \cdot)$ is strictly increasing in each of its variables. If a point M satisfies that $m_k = D_k(M)$ is a (weak) minimum of p_k for $k = \ell, r$, then M is a (weak) minimum of P .

The proof is simple: For $M' \neq M$ close enough to M we have that (m'_ℓ, m'_r) is close to (m_ℓ, m_r) , and since each m_k is a local minimum then $p_k(m'_k) \geq p_k(m_k)$. Further, as the map (D_ℓ, D_r) is injective by assumption, then $(m'_\ell, m'_r) \neq (m_\ell, m_r)$, so that $p_k(m'_k) > p_k(m_k)$ for at least one k . But since G is increasing in each variable, then $G(m'_\ell, m'_r) > G(m_\ell, m_r)$. A similar argument holds for weak minima. We note that this claim holds also with respect to the *global* minima.

We next turn our attention to minimal curves. A smooth 3D curve $C = \{C(s), s \in (0, 1)\}$ is *minimal* with respect to a potential $P(M)$ if, for each s , $C(s)$ minimizes P in the plane perpendicular to C at s . The definition is similar for a 2D curve, except that the perpendicular here is a line rather than a plane. A minimal curve can be viewed as lying at the bottom of a potential “valley”, with gradients allowed in the direction of the curve but not perpendicularly to it. Note that the definition is intrinsic, as it does not depend on the curve parameterization.

The next question to be posed is then: Consider a 3D curve C so that each of its projections c_k onto image k is a minimal curve with respect to the image potential $p_k, k = \ell, r$. Does this imply that C is minimal with respect to the combined P ? The answer here is negative. The reason is, essentially, that the projection operators do not preserve perpendicularity relations between vectors. What we do have is the following weaker property.

Claim 2. Suppose that the potential fusion function $G(\cdot, \cdot)$ is strictly increasing in each of its variables. Let C be a 3D curve so that each of its projections $c_k, k = \ell, r$ onto image k is a minimal curve with respect to the image potential p_k . Suppose further that the respective potential gradient along each curve c_k is null. Then the curve C is minimal with respect to P .

This property may be verified similarly to the point-minima claim, coupled with the uniqueness of the projections. Indeed, it immediately follows from Claim 1 that each point $C(s)$ is a weak local minimum of P . It remains to show that $C(s)$ is a *strict* minimum in the perpendicular plane. Observe first that the tangent to C at $C(s)$ is projected as a tangent to c_k at $c_k(s)$, for both $k = \ell, r$. By our injective assumption of the projection operators, no other vector can have that property. Thus, if V is a vector perpendicular to C at $C(s)$, then at least one of its projections $v_k, k = \ell, r$ will have a component perpendicular to C_k at $C_k(s)$. Thus, at least one p_k is strictly increasing in direction v_k (while the other is non-decreasing). The assumed strict monotonicity of G now implies that P is indeed strictly increasing in the perpendicular direction V .

Based on this observation, it is to be expected that when the potential gradients along the projected curves are small, then approximate minimality of the 3D curve will be maintained.

4.2. Additive vs. Multiplicative Potentials

We next compare two specific choices for the 3D potential (22). One is the basic additive potential (17). While this may be the most straightforward, other reasonable choices exist. An example is the multiplicative potential

$$P = -\alpha \sqrt{p_\ell p_r} \tag{23}$$

(the image potentials are assumed here to be negative, in accordance with (2)). Let us briefly discuss and compare the properties of the two potentials mentioned.

Figures 2 and 3 provide an illustration of these two potential functions. We use a synthetic stereo image pair of a 3D cube given in Fig. 2. The stereo configuration is the basic one described in (14), with two parallel cameras separated a distance b along the x direction. Figure 3 shows the values of the potential functions (17) and (23) over the plane $Y = 0$, as functions of

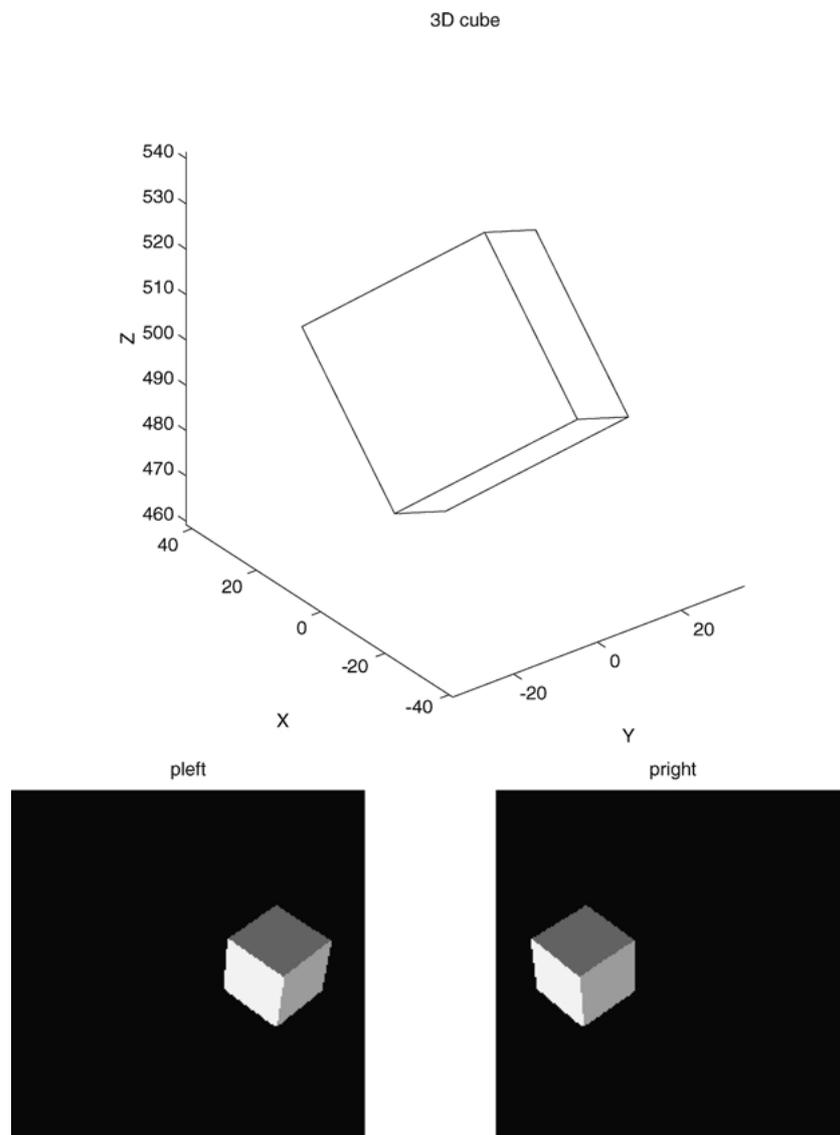


Figure 2. The synthetic 3D cube geometry (top), and the computed stereo-image pair of this cube.

the 3D coordinates X and Z . Note that the absolute value of the potential is shown, so that minima of the potential correspond to maxima in these figures; we shall refer in the following discussion to the magnitude of the potential. Both functions are seen to have four major local maxima, at the approximate (X, Z) coordinate locations $\{(-25, 500), (-10, 480), (10, 480), (25, 500)\}$, corresponding to the true locations of the four cube edges intersecting $Y = 0$. Some additional local maxima, e.g. at $(-20, 620)$, are formed through false correspondence of different edges in the two images. To compare the two potentials, consider the addi-

tive potential (17) first. According to Fig. 3, each of the four true maxima is formed by the intersection of two “ridges”, each contributed by the the potential energy term of a single image. The ridge is formed along the epipolar line intersecting the image at the object’s edge, and due to the additive nature of this potential it leads to a considerable magnitude of the overall potential (and possible local maxima) even off the intersection with the other ridge. These spurious maxima disappear in the case of the multiplicative potential function (23), as illustrated in right hand side of Fig. 3, where the potential will be small whenever one of its components

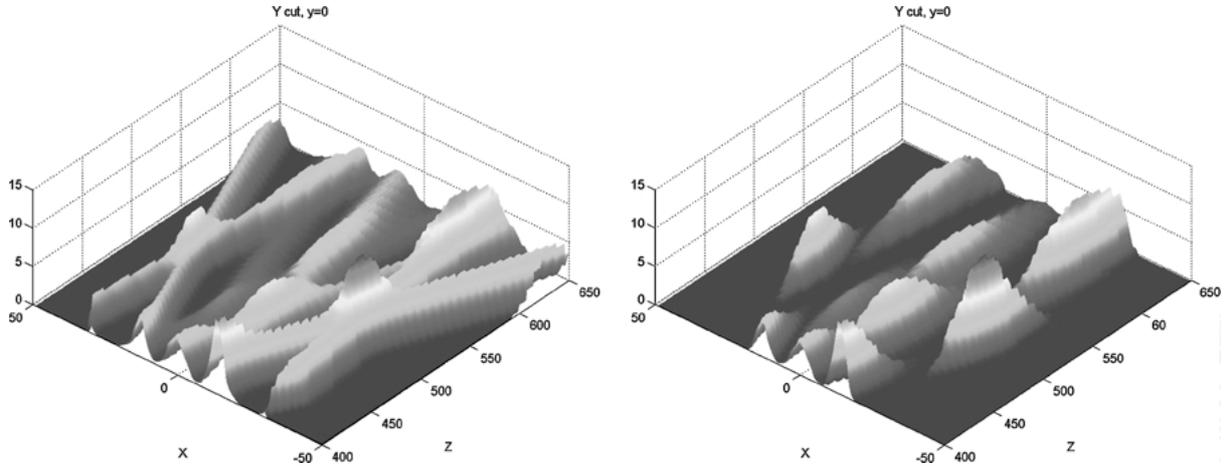


Figure 3. The additive (left) and multiplicative (right) potential functions of the stereo-pair in Fig. 2 along the coordinate $Y = 0$, as a function of the 3D coordinates X and Z . The magnitude (absolute value) of the potential is depicted.

is small. While this quenching of the potential off the intersection points seems to lead to a smoother function and possibly decrease the chances of entrapment in local extrema, the additive potential may possess some advantage when the deformable contour is far from the desired position as the information provided from one image can guide it to the correct position. Moreover, in the presence of partial occlusion in one image of the stereo-pair, the additive potential provides partial support for tracking based on the contribution of a single image to the potential field, while the multiplicative potential field is nulled. In the experiments performed so far within this research fairly close tracking was maintained, and the two potentials yielded similar performance.

4.3. Z-Axis Normalization

A major problem that was observed in the initial model simulations was the asymmetric response of the snake in the different space directions, which severely distorted the snake and degraded tracking performance. In particular, larger motions were observed along the normal (Z) direction. This effect may be attributed to the different sensitivity of the projected points to motions of the spatial point in the different directions. We will propose here a certain normalization of the potential that offsets this problem. For simplicity we relate here to the additive 3D potential (17), and the basic stereo configuration of parallel cameras, described by (14).

Recall that the 3D snake is driven by a force which equals the potential gradient (Eq. (19)). Let us relate the spatial gradients in the 3 spatial directions to the image potentials. First,

$$\begin{aligned} \nabla P(M) &= \nabla(p_\ell(D_\ell(M))) + \nabla(p_r(D_r(M))) \\ &= H_\ell(M)^T \nabla p_\ell(m_\ell) + H_r(M)^T \nabla p_r(m_r). \end{aligned} \quad (24)$$

Written explicitly, with H_k as in (13) and (14), this gives

$$\begin{aligned} \frac{\partial P}{\partial X} &= \frac{f}{Z} \left(\frac{\partial p_\ell}{\partial x_\ell} + \frac{\partial p_r}{\partial x_r} \right) \\ \frac{\partial P}{\partial Y} &= \frac{f}{Z} \left(\frac{\partial p_\ell}{\partial y_\ell} + \frac{\partial p_r}{\partial y_r} \right) \\ \frac{\partial P}{\partial Z} &= -\frac{1}{Z} \left(x_\ell \frac{\partial p_\ell}{\partial x_\ell} + x_r \frac{\partial p_r}{\partial x_r} + y \frac{\partial p_\ell}{\partial y_\ell} + y \frac{\partial p_r}{\partial y_r} \right) \end{aligned}$$

where $y = y_\ell = y_r$. Comparing the gradient in the Z direction to the other two, we see that its weighting of the image potentials is essentially different. This observation suggests normalizing the Z -direction gradient. A reasonable normalization is to multiply this gradient by a position-dependent factor of magnitude

$$K_Z(M) = \frac{f}{\frac{1}{2}(|x_\ell| + |x_r|) + |y|}. \quad (25)$$

This normalization proved very effective in solving the Z -distortion of the contour, and was implemented in all the experiments reported in the sequel.

We observe that the normalized gradient will not in general be a true gradient of any potential function. This is of course not a problem, since the potential only serves to define the driving forces in the model through its gradients.

4.4. Best-Fit Projections

In this subsection we derive a formula for the “potential-gradient” (namely, the driving force for the 3D snake) which relies on some optimal-fit considerations. The resulting force can be viewed as a normalized additive-potential term.

Recall that the term ∇P (with minus sign) in (19) is in effect a spatial force field that drives the 3D snake. Assume for the moment that ∇P is an actual 3D gradient; it would then be reasonable to require its projections on the image planes to coincide with the respective image gradients. That is:

$$\begin{aligned} H_\ell(M)\nabla P(M) &\cong \nabla p_\ell(m_\ell), \\ H_r(M)\nabla P(M) &\cong \nabla p_r(m_r), \end{aligned}$$

where, as usual, $m_k = D_k(M)$. That will be the starting point for the following derivation. We make two observations. First, we shall not insist here (as we did not in the previous subsection) that the driving force field will be integrable (i.e. an actual gradient of some scalar function). We therefore replace ∇P in (19) and in the last equations with F , a general force field. Next, we observe that the last two equations cannot in general be satisfied simultaneously. Algebraically, at each point M we have four scalar equations for the three components of F . Indeed, a pair of vectors in stereo images can be the projections of a single spatial vector only if they satisfy certain geometric (epipolar) constraints. We thus define F as the optimal least-squares approximation to these equations, namely $F(M)$ is the solution of the following linear least-squares problem:

$$\begin{aligned} \min_F \{ &\|H_\ell(M)F - \nabla p_\ell(m_\ell)\|^2 \\ &+ \|H_r(M)F - \nabla p_r(m_r)\|^2 \}. \end{aligned}$$

The solution is

$$\begin{aligned} F(M) &= (H_\ell^T H_\ell + H_r^T H_r)^{-1}(M) \\ &\times (H_\ell^T(M)\nabla p_\ell(m_\ell) + H_r^T(M)\nabla p_r(m_r)). \end{aligned}$$

We may now compare this expression with the force field induced by the additive potential (17), as

computed in (24). It is evident that the expressions differ by the inverse matrix $(H_\ell^T H_\ell + H_r^T H_r)^{-1}$, which can be considered a multiplicative normalization of the additive potential force.

To get a quantitative idea of this normalization matrix, we evaluate it for the basic stereo configuration defined in (14). By straightforward computation and some re-arranging,

$$\begin{aligned} &(H_\ell^T H_\ell + H_r^T H_r)^{-1}(M) \\ &= \frac{Z^2}{2d_x^2} \begin{bmatrix} \bar{x}^2 + d_x^2 & \bar{x}y & \bar{x} \\ \bar{x}y & y^2 + d_x^2 & y \\ \bar{x} & y & 1 \end{bmatrix} \end{aligned}$$

where $\bar{x} = \frac{1}{2}(x_\ell + x_r)$, $d_x = \frac{1}{2}(x_\ell - x_r)$, and $y = y_\ell = y_r$. It is interesting to note that the elements of the matrix depend only on image coordinates. If we compare the diagonal elements to the Z -axis normalization suggested in (25), one may observe some similarity in the relative normalization of the Z axis to the X and Y axes; however, here we have different normalizations of the X and Y axes, and the dependence on the image coordinates is quadratic rather than linear. An empirical comparison of different normalization schemes has yet to be performed.

5. Space and Time Discretization

We briefly describe next the discretization in space and time that will be used in the numerical implementation of the 3D velocity snake. The continuous model (19) is transformed into linear discrete-time equations with nonlinear image-based inputs through conventional finite difference techniques.

Consider first the space discretization of the equations of motion. Let $U = [u_1, \dots, u_N]$ be an equidistant sampling of the contour $C(s)$, with $u_k = C(s_k)$ and $s_k = k/N$. We use the standard forward Euler approximation, $C_s(s_k) = N^{-1}(u_{k+1} - u_k)$ and $C_{ss}(s_k) = N^{-2}(u_{k+1} - 2u_k + u_{k-1})$. Substituting in (21) and (19), we obtain the discrete-space version of the equations of motion:

$$\begin{aligned} MU_{tt} + \gamma &(H_\ell^T \nabla I_\ell (\nabla I_\ell^T H_\ell U_t + (I_\ell)_t)) \\ &+ H_r^T \nabla I_r (\nabla I_r^T H_r U_t + (I_r)_t) \\ &+ \beta DD^T U_t + KU = -\nabla P(U) \end{aligned} \quad (26)$$

where K is a deformation matrix defined in Terzopoulos and Szeliski (1992), and M is a diagonal

mass matrix (the unit matrix in our case), and D is the difference matrix

$$D = \begin{bmatrix} -1 & 1 & 0 & & 0 \\ 0 & -1 & 1 & \ddots & \\ \vdots & 0 & \ddots & \ddots & 0 \\ 0 & & \ddots & & 1 \\ 1 & 0 & \dots & 0 & -1 \end{bmatrix}. \quad (27)$$

The temporal discretization of (26) is again performed by straightforward central difference approximation. We shall omit further details here. The following remarks are however in order regarding the discretization and implementation scheme:

1. It should be stressed that the discrete time step in the snake model is taken as the video inter-frame interval. This means that exactly one iteration is performed for each stereo pair. In this iteration the velocity term is fully employed in order to place the contour at the position projected by the apparent motion calculation, while the other terms account for attraction to image features and rigidity constraints of the snake.
2. As an option, additional iterations per frame may be used in order to improve the accuracy of correspondence between the snake and image features. However, these must be performed for the basic model without the velocity terms (which are fully utilized in the first iteration). Thus, while the first iteration uses (26), additional iteration may rely on the basic snake model of the form (Terzopoulos and Szeliski, 1992)

$$MU_{tt} + BU_t + KU = -\nabla P(U) \quad (28)$$

where B is an appropriate damping matrix.

3. We note that contour space discretization (in terms of the contour parameter) is held fixed across images. This leads to *linear* update equations, which are relatively robust with respect to large time steps as performed here. (In comparison, in the geodesic snake model the curve is in effect re-parameterized in each step according to arc-length, which leads to non-linear evolution equations. As noted in Goldenberg et al., 2001, these equations require a small numerical step for stability.)
4. Keeping a fixed space parameterization may cause performance degradation due to such phenomena as

point bunching. This has indeed been observed in our experiments in the absence of the velocity term. However, the incorporation of this term practically eliminates these problems for the duration of our experiments, as reported next.

6. Experimental Results

The performance of the proposed three-dimensional velocity snake is demonstrated in this section by applying it to simulated and real image sequences. It should be emphasized that these experiments are intended to demonstrate the applicability of the velocity snake (with a single iteration per frame) to 3D tracking at video rate, and to demonstrate the improvement achieved by the velocity term. No attempt was made to deal with hard background and clutter problems. Accordingly, the basic edge-seeking potential of the form (2) was used in our experiments.

We first show the tracking capability of the three-dimensional velocity snake (26), applied to a synthetic cube sequence, and compare the results to that of the three-dimensional active contour (28) which does not employ the velocity term. As was noted, the latter is a direct generalization of the original snake model of Kass et al. (1987) to the three-dimensional space. The tracking capability is then demonstrated on three real stereo-image sequences, two with rigid motion (book and chair) and one with nonrigid motion (a human hand).

Prior to the calculation of the image gradients and derivatives, the image sequences were smoothed in space by a Gaussian filter with $\sigma = 2$ pixels for the potential computation, and with $\sigma = 5$ for the velocity measurements. This choice of smoothing parameters was related to the maximal expected image motion. The position derivatives in the image planes were calculated by a 3×3 Prewitt operator, and the time derivative by simple differences between successive images.

The results presented in this section are all for the active contour models (26) or (28) with the additive potential function (17), normalized according to the Z -axis normalization described in Section 4.3. The image potential is $p = -k\|G_\sigma * \nabla I\|$. The parameters of the contour models were chosen empirically. For an approach to self tuning of the parameters in similar models see Blake and Isard (1998).

Similar experiments were carried out using the multiplicative potential (17), both for the synthetic cube and the real images (book and chair sequences). In all

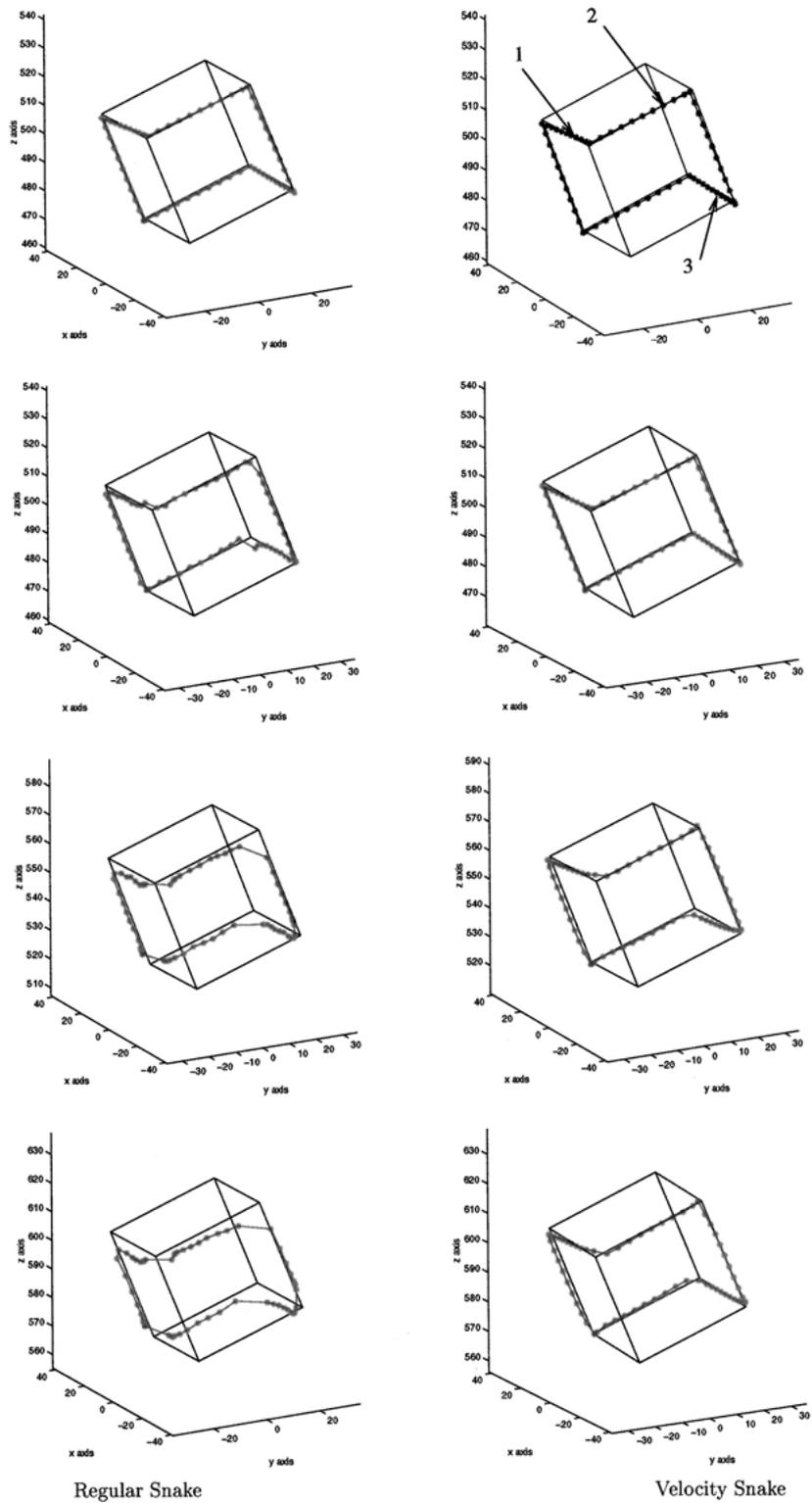


Figure 4. Tracking results of the synthetic cube with the regular snake (28) and velocity snake (26) at the initial position (top), and at the time of the 1st, 50th and 97th frame (bottom). Note that the cube is in motion as the Z coordinate values are increasing.

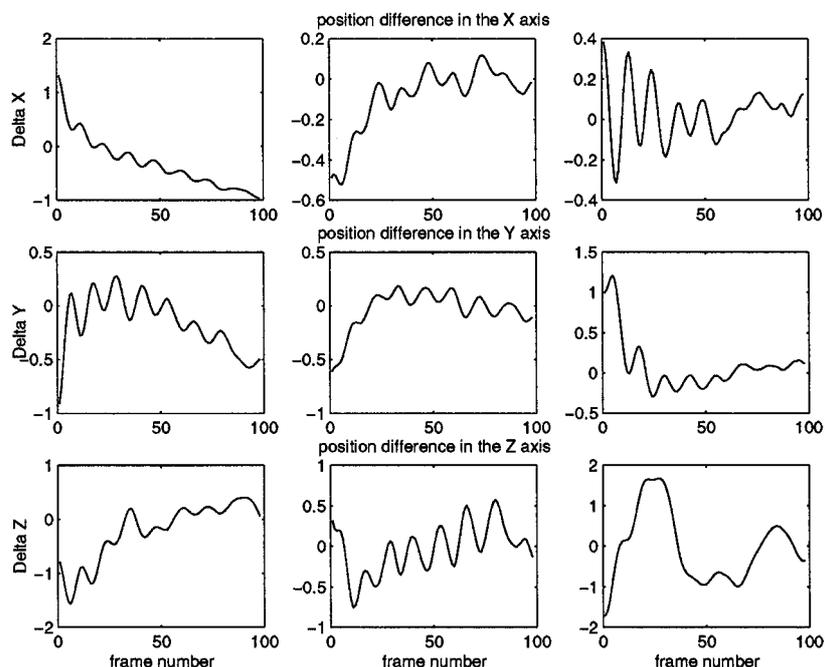


Figure 5. Position tracking errors for three snake points relative the synthetic cube. The errors in the three spatial axes are shown for the entire 97 frame simulation, point 1 is on the left.

cases the results were similar to the ones obtained with the additive potential, and are therefore omitted here.

For the regular three-dimensional snake (28), the contour position at each sampled image was computed based on 100 iterations. This number was chosen in order to allow the snake a reasonable time to stabilize on a solution. This heavy computation is in contrast to the reduced complexity provided by the velocity snake (26) which requires only one iteration per frame, and provides estimation of both shape and motion. We note that using the basic model (28) with a single iteration per frame resulted in an immediate loss of tracking.

Synthetic Cube: In order to fully monitor the tracking performance of the proposed 3D active contour under controlled conditions, the model was applied to a synthetic stereo image sequence of a simulated cube (Fig. 2). We simulated a three-dimensional cube with dimensions $40 \times 40 \times 40$ length units, moving in the positive Z direction (away from the camera) with a velocity of 25 length units per second. The cube was initially positioned at a distance of 500 units in the Z direction. The two-dimensional stereo image sequence was formed by using perspective projection with a camera focus of $f = 500$ pixels and with a gray-scale of 64 levels. Each visible cube face was painted with a

different color (gray-scale level of 15, 40 and 63 respectively) with the background painted in black (gray-scale level of 1). The image sequence was made up of 97 images, sampled at a rate of 25 frames per second ($T = 1/25$ sec.).

Stereo Configuration: Both the synthetic and the real-image tracking experiments used the basic stereo configuration described in (14): two identical cameras with identical orientation, with a separation b along the x axis. The baseline b was chosen small enough to prevent qualitative shape distortions between the two views, $b = 100$ was used in the cube experiment.

Contour Parameters: We used contour models with a three-dimensional spatial sampling distance of 4 units. The model (28) was used with $\mu = 1$, $w_1 = 5$, $w_2 = 0.1$ and $\gamma = 100$ and the three-dimensional velocity snake with $\mu = 1$, $w_1 = 5$, $w_2 = 0.01$, $\gamma = 0.01$ and $\beta = 1000$. In addition, the potential was multiplied by a factor of 100 in order to allow the image to form the proper attraction force.

Tracking Results: The results of tracking the moving cube are shown in Fig. 4. We show samples of the tracking results of the regular snake (28) and of the velocity snake (26), at the initial frame, and at the 1st, the 50th and the 97th frame, respectively. It can be seen that both models successfully track the cube throughout

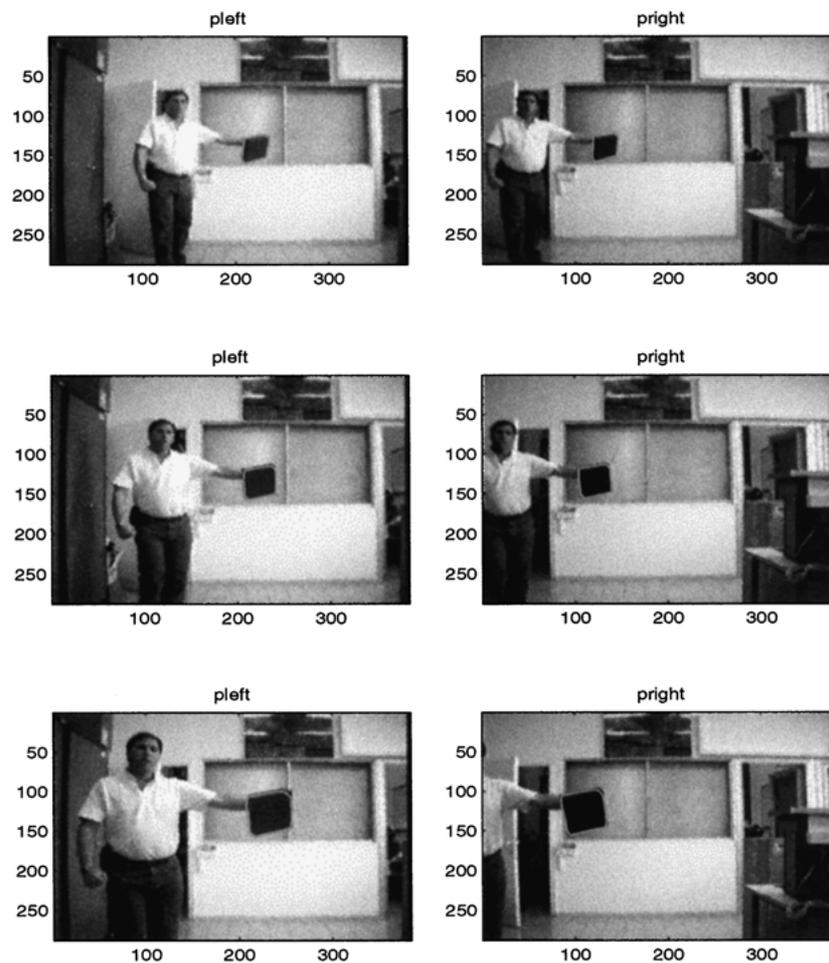


Figure 6. Samples of three stereo images (1st frame, 51th frame and the 101th frame) with the position of the three-dimensional velocity snake projected onto them.

the sequence. However, the tracking performance of the regular three-dimensional snake exhibits two major problems. The first concern is with the bunching of the sampling points towards the center of the cube sides. This occurs since there is no correspondence information between points in successive images. The second is that the contour lags behind the synthetic cube. Both of these issues are rectified when velocity term is added to the three-dimensional snake. This term gives an estimate of the object velocity vector between subsequent frames which prevents the bunching of the snake points and improves the tracking accuracy.

The tracking errors for three representative snake points are depicted in Fig. 5. The points were chosen on different edges of the cube, and are indicated on the upper-right image in Fig. 4. For each of these

snake points, we match a fixed reference point on the cube edge, which is chosen as the closest edge point in the initial frame (the initial placement of the snake contained some deliberate error). The tracking error at each frame is defined as the difference (in each spatial axis) between the snake point and the current position of its matched cube point. It may be seen that the tracking errors are in the order of one unit or less. (Note that the perspective projections (11) with $f = 500$ and $Z \approx 500$ imply that a spatial error of one unit corresponds to one image pixel.) In particular, it may be seen that each point closely maintains its relative position to the original cube point, which is indicative of the lack of point bunching. Only in the X coordinate of the first point, one can observe a persistent increase in the error, which reflects a gradual sliding of this point along the

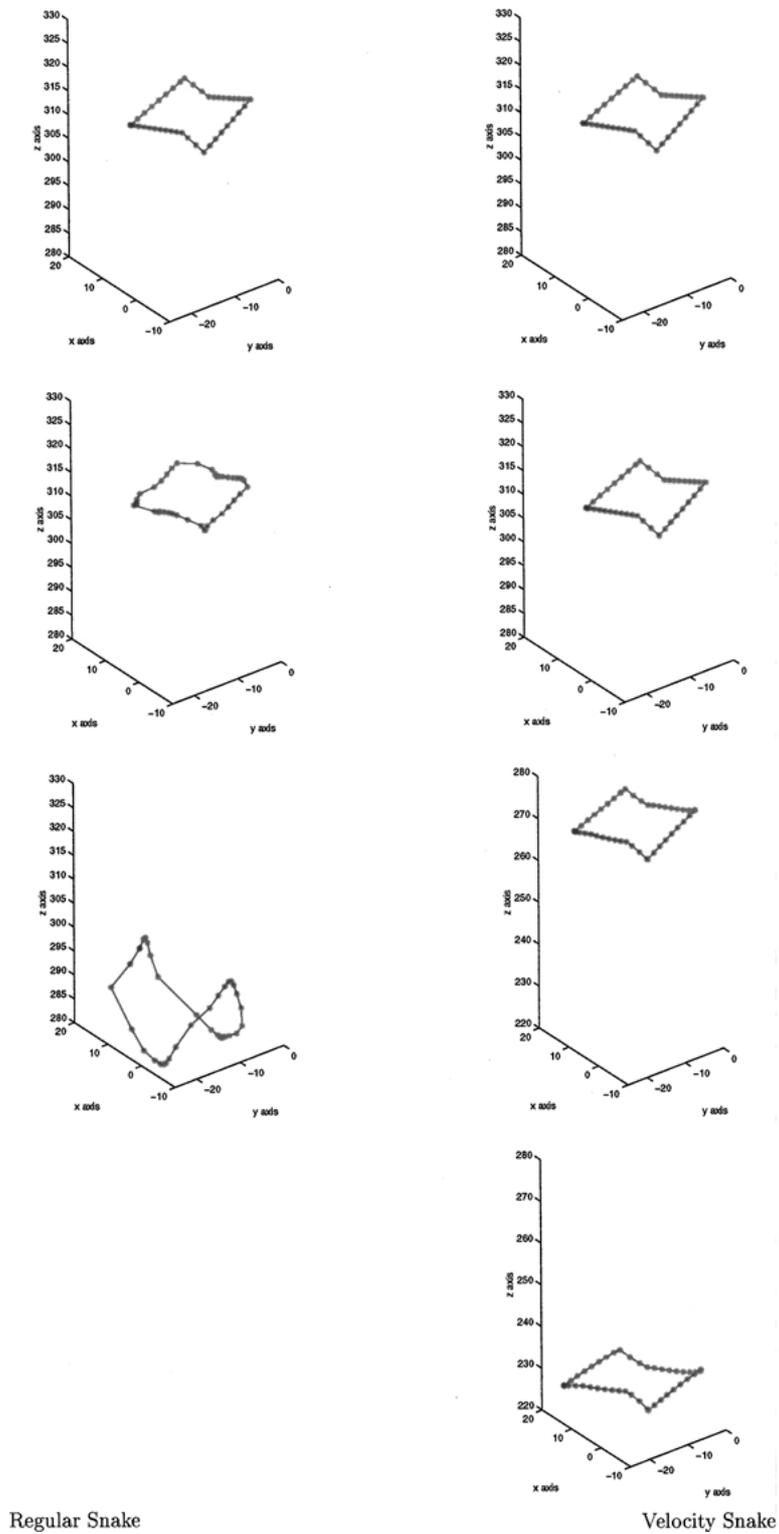


Figure 7. Tracking results of the book in Fig. 6 with the velocity snake (26) and with the regular snake (28) at the initial frame (top), and at the 1st, the 50th and the 101th frame, respectively.

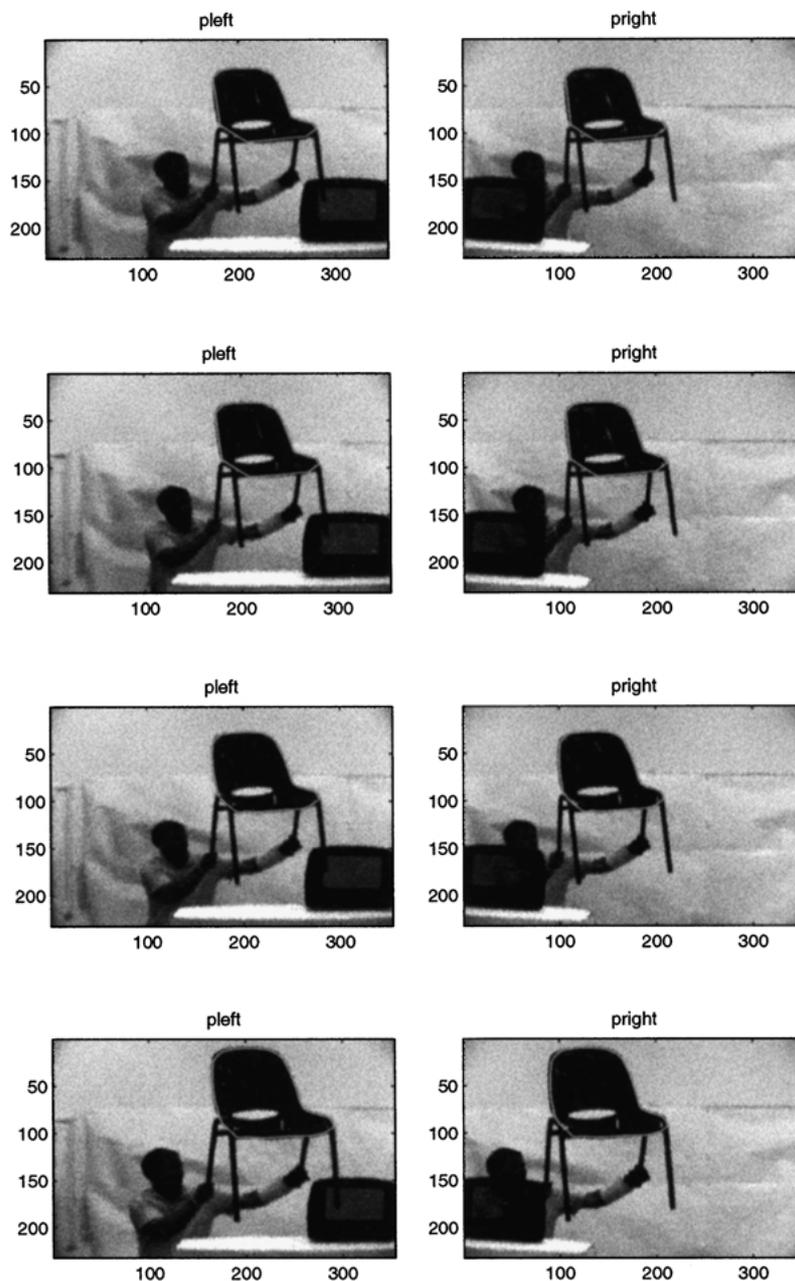


Figure 8. Samples of the chair stereo sequence (initial position, 1st frame, 51th frame and the 96th frame) with the position of the three-dimensional velocity snake projected onto them.

cube edge. It may also be seen that the errors are oscillatory in nature, these oscillations may be damped (if needed) by the use of an appropriate temporal smoothing filter, such as the Kalman filter.

Book Tracking: Next we present the results of tracking a book in office background with the regular model

(28) and with the velocity snake (26). The book measured of $24 \times 21 \times 6$ cm, and was moving towards the camera (Z direction) from a distance of 3.1 meters to 2.3 meters. The image sequence comprised of 101 images for the three-dimensional velocity snake. The stereo images were captured using a stereo rig

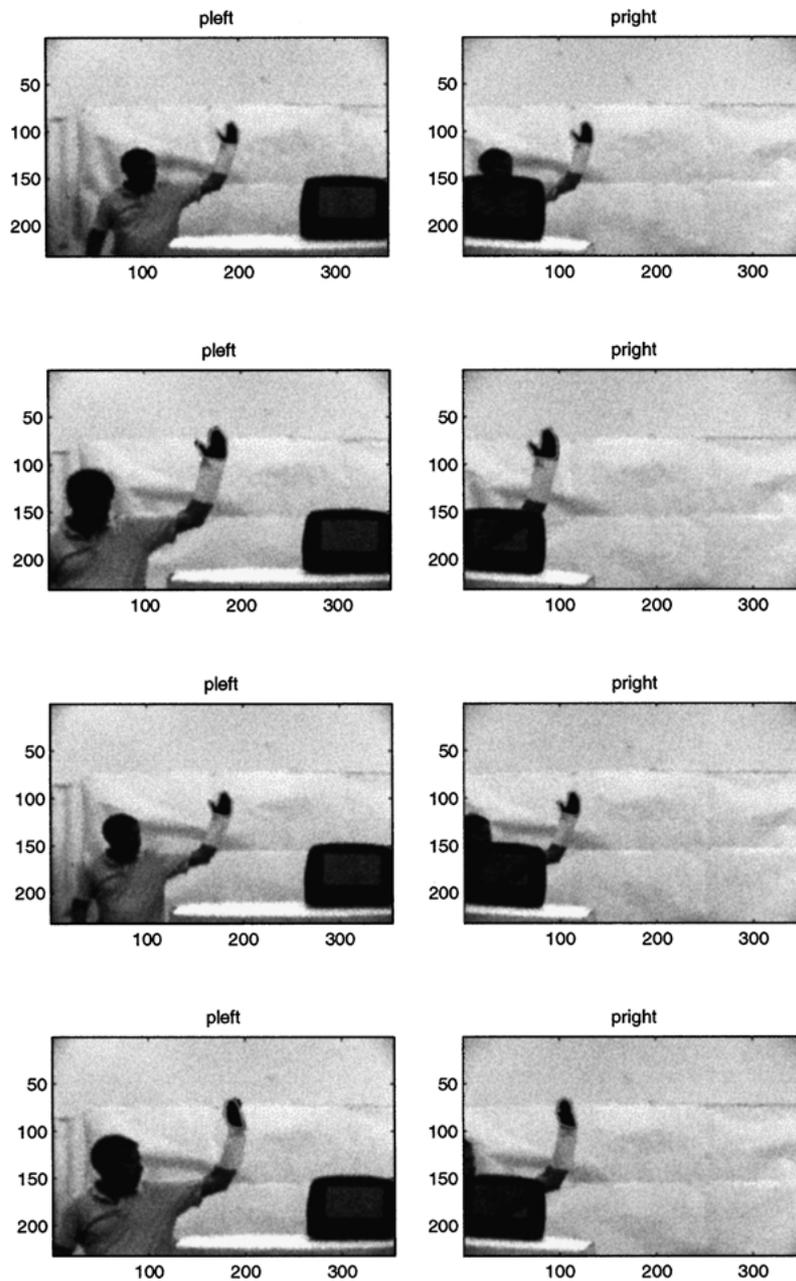


Figure 9. Samples of the hand stereo sequence (initial position, 100th frame, 225th frame, and 345th frame) with the position of the three-dimensional snake projected onto them.

in our standard configuration—two identical synchronized cameras positioned $b = 58$ cm apart with parallel orientation. The focal length of each camera was $f = 476$ pixels. The images had a gray-scale of 256 colors and were digitized on two Silicon Graphic workstations at a rate of 25 frames per second.

Contour Parameters: We used the contour models with a three-dimensional sampling distance of 2.5 cm. The model (28) was used with $\mu = 1$, $w1 = 5$, $w2 = 0.01$ and $\gamma = 10$. For the three-dimensional velocity snake, we used $\mu = 1$, $w1 = 1$, $w2 = 0.01$, $\gamma = 0.001$ and $\beta = 2000$. For the regular three-dimensional

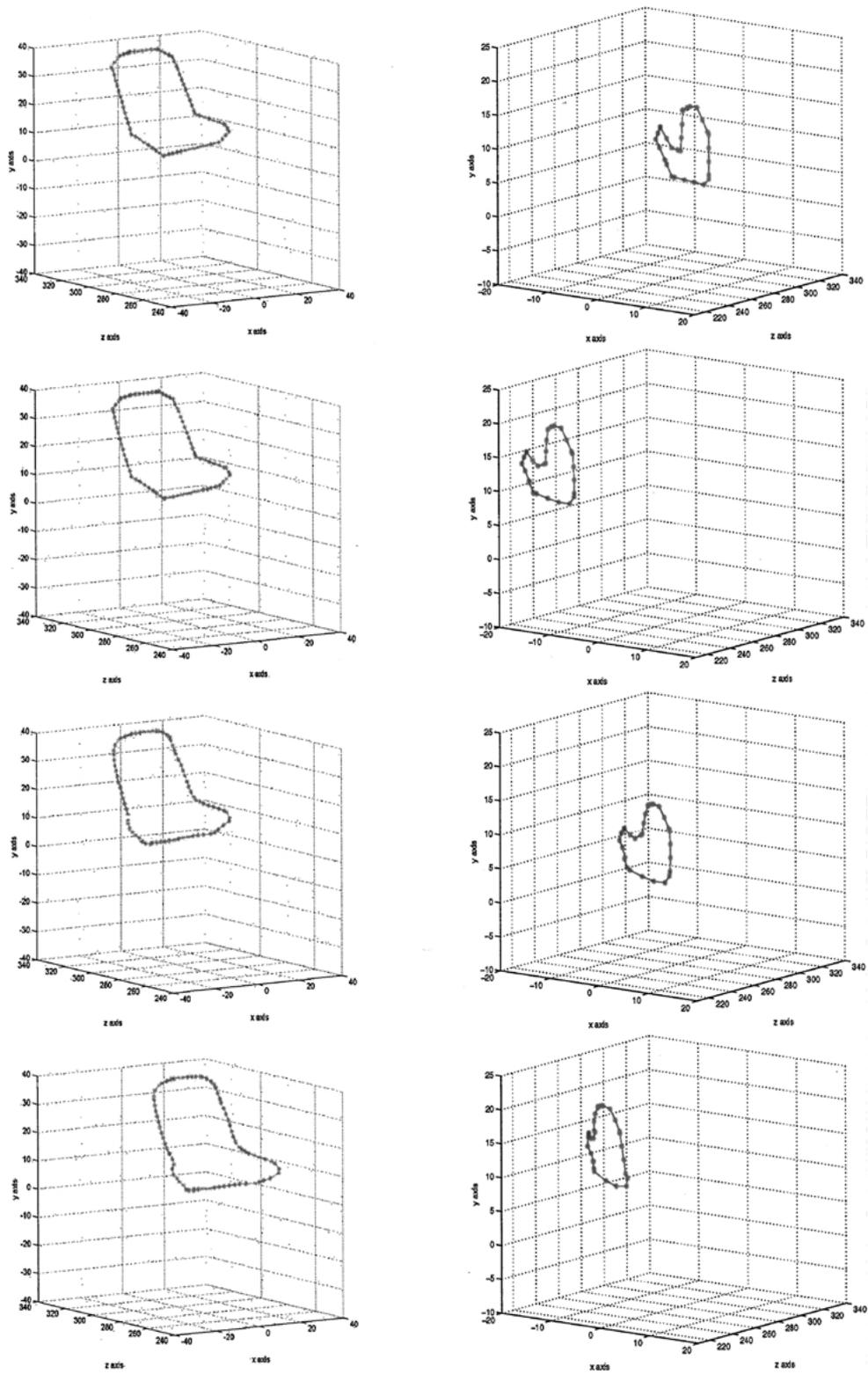


Figure 10. The 3D velocity snake position for the images of Fig. 8 (left) and of Fig. 9 (right).

snake, the potential was multiplied by a factor of 10, while a factor of 5 was used for the three-dimensional velocity snake.

Tracking Results: The results of tracking of the book in 3D and the corresponding projection onto the 2D stereo images, are shown in Figs. 6 and 7. We show the results of the model (28) only until the 51th frame as they were very poor following that frame. The velocity snake gave precise results along the entire 101 images of the sequence.

Chair Tracking: In Figs. 8 and 10 we show the tracking results of a “flying” chair with the proposed 3D contour model. The sequence, composed of 100 images, illustrates the tracking capability in the presence of non-constant object velocity. Note that the proposed model does not employ the rigidity property of the object in 3D. The only information it uses is the constraints on the dynamical behavior of shape and motion along time, which is embedded within the parameters of the tracking contour. The results in Figs. 8 and 10 demonstrate high quality of tracking in the presence of non-constant motion.

Hand Tracking: In Figs. 9 and 10 we depict the tracking results of shape and of motion of the nonrigid contour of a moving hand. The results were obtained by the velocity snake (26). In this sequence we aimed to demonstrate the robustness of the three-dimensional velocity snake in a long image sequence (about 12 seconds) which included changes in velocity of the object being tracked (forward and backward motion) as well as a change in the object’s shape and the presence of occlusions (the hand turns sideways at the end of the sequence). The sequence was comprised of 345 images of a hand moving towards and away from the cameras from a distance of 3.4 meters to 2.4 meters. The hand moved backwards and forwards as well as to the sides, with close tracking maintained throughout the sequence. A noticeable increase in the tracking error was observed when the motion changed directions, but snake quickly caught up to the object. This sequence also demonstrates the three-dimensional velocity snake’s capability of tracking changes in shape of the three-dimensional object.

7. Concluding Remarks

We have considered in this paper the problem of 3D tracking of deformable contours in stereo movies. Our approach relies on the explicit representation and evolution of the active contour in three-dimensional space,

under the influence of a spatial force field induced by an appropriate combination of image potentials. This approach may be contrasted with the option of using separate 2D contours in each of the images, and forming the 3D contour at each step through stereo correspondence. The proposed approach offers important advantages as the actual problem geometry is retained. Structural constraints and prior knowledge regarding the size and shape of the tracked object may be directly incorporated. The same holds for the compensation of known camera motion and the incorporation of geometric constraints from the environment. Data from other 3D sensors, if available, may be easily integrated. The correspondence problem is practically eliminated once the contour is initialized; in particular, such phenomena as self-intersecting of the 2D contours should not affect the 3D contour evolution. Occlusions may also be better handled as the position relative to possible occluding objects may be computed, and the contour maintains its spatial state and shape in the absence of image-induced forces.

The proposed tracking model incorporates within the 3D active contour a velocity term related to the optical flow in the two images. We introduced a particular choice of the relevant parameters, which greatly reduces the computational requirements. The experimental results indicate successful tracking of simulated and real scenes and clearly demonstrate the performance enhancement associated with the velocity term.

The results of this paper demonstrate the feasibility of the proposed approach for real time 3D contour tracking. Additional work on the basic model is required in order to optimize the computational load and improve numerical stability. Clearly, to obtain state-of-the-art tracking performance one needs to incorporate additional elements, in particular more elaborate image potentials, and additional shape and motion constraints when available. The development of similar ideas in the context of geometric snake models and related level set approaches are interesting areas for additional research.

Acknowledgments

The authors would like to thank the referees for their useful comments, which contributed to the presentation and led to further elucidation of several important issues. Funding for this work was provided, in part, by the Materials and Engineering Physics Program of the Office of Basic Energy Sciences, U.S. Department

of Energy, through the Center for Engineering Science Advanced Research, Oak Ridge National Laboratory. The Oak Ridge National Laboratory is managed by UT-Battell, LLC, for the Department of Energy, under contract No. DE-AC00OR22752.

References

- Bascle, B., Bouthemy, P., Deriche, R., and Meyer, F. 1994. Tracking complex primitives in an image sequence. In *Proc. 12th IAPR International Conference on Pattern Recognition (ICPR'94)*, Jerusalem, pp. 426–431.
- Bascle, B. and Deriche, R. 1993a. Energy-based methods for 2D curve tracking, reconstruction and refinement of 3D curves and applications. In *Geometric Methods in Computer Vision, SPIE Vol. 2031*, B.C. Vemuri (Eds.), pp. 282–293.
- Bascle, B. and Deriche, R. 1993b. Stereo matching, reconstruction and refinement of 3D curves using deformable contours. In *Proc. IEEE Int. Conf. on Computer Vision*, Berlin, pp. 421–430.
- Berger, M.-O., Mozelle, G., and Laprie, Y. 1995. Cooperation of active contours and optical flow for tongue tracking in X-ray motion pictures. In *Proc. 9th Scandinavian Conference on Image Analysis*, Puerto Rico, pp. 1094–1099.
- Bertalmio, M., Sapiro, G., and Randall, M.G. 2000. Morphing active contours. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(7):733–737.
- Blake, A. and Isard, M. 1998. *Active Contours*. Springer: London.
- Blake, A. and Yuille, A. 1992. *Active Vision*. MIT Press: Cambridge, MA.
- Caselles, B. and Coll, B. 1996. Snakes in movement. *SIAM Journal on Numerical Analysis*, 33:2445–2456.
- Caselles, V., Kimmel, R., and Sapiro, G. 1997. Geodesic active contours. *International Journal of Computer Vision*, 22(1):61–79.
- Caselles, V., Kimmel, R., Sapiro, G., and Sbert, C. 1997. Minimal surfaces based object segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(4):394–398.
- Cham, T. and Cipolla, R. 1997. Stereo coupled active contours. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, Puerto Rico, pp. 1094–1099.
- Deriche, R., Bouvin, S., and Faugeras, O. 1998. Front propagation and level-set approach for geodesic active stereovision. In *Third Asian Conference On Computer Vision*, Hong Kong.
- Faugeras, O. and Keriven, R. 1999. Variational principles, surface evolution, PDE's, level set methods and the stereo problem. *IEEE Transactions on Image Processing*, 7(3):336–344.
- Goldenberg, R., Kimmel, R., Rivlin, E., and Rudzsky, M. 2001. Fast geodesic active contours. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 10(10):1467–1475.
- Horn, B.K.P. 1986. *Robot Vision*. MIT Press: Cambridge, MA.
- Jain, A.K., Zhong, Y., and Dubuisson-Jolly, M. 1998. Deformable template models: A review. *Signal Processing*, 71:109–129.
- Kass, M., Witkin, A., and Terzopoulos, D. 1987. Snakes: Active contour models. *International Journal of Computer Vision*, 1(4):321–331.
- Kichenassami, S., Kumar, A., Olver, P., Tannenbaum, A., and Yezzi, A. 1996. Conformal curvature flows: From phase transitions to active vision. *Arch. Rational Mech. Anal.*, 134:275–301.
- McInerney, T. and Terzopoulos, D. 1996. Deformable models in medical image analysis: A survey. *Medical Image Analysis*, 1(2):91–108.
- Paragios, N. and Deriche, R. 1999. Geodesic active regions for motion estimation and tracking. In *Proc. 7th IEEE Int. Conf. on Computer Vision*, Kerkyra, Greece, pp. 688–694.
- Paragios, N. and Deriche, R. 2000. Geodesic active contours and level sets for the detection and tracking of moving objects. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(3):266–280.
- Peterfreund, N. 1997. The velocity snake. In *Proc. IEEE Non Rigid and Articulated Motion Workshop*, Los Alamitos, CA, pp. 70–79.
- Peterfreund, N. 1999. The velocity snake: Deformable contour for tracking in spatio-velocity space. *Computer Vision and Image Understanding*, 73(3):346–356.
- Sapiro, G. 2001. *Geometric Partial Differential Equations and Image Analysis*. Cambridge University Press, New York.
- Terzopoulos, D. and Szeliski, R. 1992. Tracking with kalman snakes. In *Active Vision*, A. Blake and A. Yuille (Eds.), MIT Press: Cambridge, MA.
- Terzopoulos, D., Witkin, A., and Kass, M. 1988. Constraints on deformable models: Recovering 3D shape and nonrigid motion. *Artificial Intelligence*, 36:91–123.