

Shadow resistant tracking using inertia constraints

Hao Jiang, Mark S. Drew*

School of Computing Science, Simon Fraser University, Vancouver, BC, Canada V5A 1S6

Received 30 July 2004; received in revised form 25 April 2005; accepted 1 September 2005

Abstract

In this paper, we present a new method for tracking objects with shadows. Traditional motion-based tracking schemes cannot usually distinguish the shadow from the object itself, and this results in a falsely captured object shape. If we want to utilize the object's shape information for a pattern recognition task, this poses a severe difficulty. In this paper we present a color processing scheme to project the image into an illumination invariant space such that the shadow's effect is greatly attenuated. The optical flow in this projected image together with the original image is used as a reference for object tracking so that we can extract the real object shape in the tracking process. We present a modified snake model for general video object tracking. Two new external forces are introduced into the snake equation based on the predictive contour and a new chordal string shape descriptor such that the active contour is attracted to a shape similar to the one in the previous video frame. The proposed method can deal with the problem of an object's ceasing movement temporarily, and can also avoid the problem of the snake tracking into the object interior. Global affine motion estimation is applied to mitigate the effect of camera motion, and hence the method can be applied in a general video environment. Experimental results show that the proposed method can track the real object even if there is strong shadow influence.

© 2005 Pattern Recognition Society. Published by Elsevier Ltd. All rights reserved.

Keywords: Shadows; Tracking; Snakes; Active contours; Variational; Illumination invariance; Inertia

1. Introduction

Shadows present a confounding factor for correct object tracking. Traditional motion detection schemes cannot distinguish the moving object and the shadows moving with it. Therefore, object tracking results based on traditional schemes usually produce contours based on a combination of the object and its shadow. This kind of result will pose severe difficulties if the contour is further passed to an analyzer for object recognition. Eliminating the shadow and tracking the real contour of an object is a challenging problem. Different schemes have been presented to try to attenuate the shadow's influence in applications such as object tracking and still image segmentation. In Ref. [1] the shadow detection and elimination problem was studied in the context of road surveillance. In the specific application, lighting

conditions were restricted to sunlight, the internal and external parameters of the video camera were fixed, and the target object was restricted to a walking human being. A method was presented for locating the real position of a walking human by extracting the core lines of the human and the core lines of the shadows based on a motion detection map. In Ref. [2] a geometrical scheme based on stereo vision was presented for shadow elimination in surveillance video tracking. The scheme is based on image subtraction, with the image captured by one camera first projected onto the road plane and then further projected onto the image plane of the second camera. The road maps of two images should map perfectly while other parts such as walking humans will not map well. The difference of the two images, thresholded by a given value, yields a mask eliminating everything on the road plane including the moving shadows. In Ref. [3], simple illumination invariant features were applied to obtain an image, which apprehends differences between surface materials. Since cast shadows only change the illumination of backgrounds, the illumination invariant features will

* Corresponding author. Tel.: +1 604 291 4682; fax: +1 604 291 3045.

E-mail addresses: hjiangb@cs.sfu.ca (H. Jiang), mark@cs.sfu.ca (M.S. Drew).

attenuate shadow effects. The method is applied in the context of still image segmentation. In Ref. [4], a statistics method is presented for pixel classification. The features used include the luminance and normalized chrominance vector. The color change of a pixel is described by multiplying each color channel by a constant, via a diagonal model of color change. Pixels are classified into three classes: background, foreground, and shadow, based on maximum a posteriori classification. Spatial information is also applied to improve the dense region classification result. In Ref. [5], the shadow detection problem is studied based on a model similar to the Phong model. Heuristic methods are presented to classify the shadow and foreground object. A recent shadow detection scheme designed for outdoor scenes is presented in Ref. [6].

In this paper, we present a different method, based on a physics-based illumination invariant color space, and on an inertia-enhanced snake model, for reliable object tracking in a general video environment. If lighting is approximately Planckian, then in Wien's approximation the resulting simple exponential form of the illumination spectrum leads to the conclusion that as temperature T changes, characterizing the illumination color, a log–log plot of two-dimensional $\{\log(R/G), \log(B/G)\}$ values for any single surface forms a straight line provided camera sensors are fairly narrow-band [7–9]. Thus, lighting change reduces to a linear transformation along an almost straight line, even for real data with only approximately Planckian lighting. For a target with many paint patches, mean-subtracted log–log plots all cluster around a single line through the origin that characterizes lighting change. The invariant image is thus the gray-scale image that results from projecting log–log pixel values onto the direction orthogonal to lighting change, within and outside the umbra; the projection greatly attenuates shadowing.

Based on this color projection, we further present an inertia-enhanced snake model for tracking objects with shadows. We devise two inertia terms. The first term is based on the predictive contour, and the second is based on a new chordal shape descriptor. These two additional terms force the active contour to converge to a shape similar to the one in the previous video frame. The inertia energy term makes the snake ignore distracting elements, and thus no precise initial contour is needed. Moreover, if the object stops moving temporarily, the snake will evolve according to the inertia term in the predictive contour and chordal constraint term and converge to a similar shape to the previous frame and also correspond to the motion prediction result. We adopt an affine motion model for global motion estimation and camera motion compensation with the result that our scheme can work in a general video environment. Comparing to other standard contour tracking schemes such as [10–12], the proposed scheme does not need a training process. The complexity of the algorithm is comparable to the standard snake and is thus suited for real-time applications.

The organization of the paper is as follows. We first study shadow-invariant image space in Section 2. We show that under Planckian lighting, the log–log plot of ratios $(\log(R/G), \log(B/G))$ forms a straight line for each material, for narrow-band sensor cameras. Based on this observation, we set out a camera calibration scheme for shadow-invariant image generation in Section 2.1. In Section 3, we present the tracking scheme based on an inertia snake model and shadow-invariant image for shadow resistant video tracking. The modified snake equation is studied in Section 3.1. Contour prediction based on iterative conditional modes (ICM) is presented in Section 3.2. In Section 3.3, we show how to generate external forces based on global motion compensated motion detection and gradient vector flow. The numerical scheme for the proposed snake equation is presented in Section 3.5, and the tracking system is presented in Section 4. Experiments, results, and discussions are presented in Section 5.

2. Shadow-invariant image space

Shadows are usually classified as self-shadows and cast shadows. Self-shadows result from part of the object blocking some light from another part of the same object. Self-shadows usually pose a minor problem for tracking tasks. However, cast shadows, caused by one object shadowing another object in the scene, can be caused by the background objects shadowing the tracking target or the target's own shadow on the background object. We are most interested in cast shadows, especially the shadow moving along with the target object. It is well known that the basic difference between a shadow area and its surrounding non-shadow area is due to lighting change. Since shadows are caused by illumination change, color constancy methods present an appropriate method for shadow detection and elimination. By using an illumination-color and intensity *invariant* scheme, we extract the quantity characterized by the object's surface reflectance property only, and thus eliminate the influence of illumination change entirely. The difficulty caused by shadows in motion detection and object shadow classification can thus be solved by applying a particular type of color constancy scheme to obtain an *invariant* image.

In this section, we present a color constancy scheme based on the assumption of Planckian lighting and approximately delta-function sensors and show how to generate the invariant image. Planckian lighting is a good approximation for a wide range of lighting sources, such as the sun and many indoor illuminants. The seemingly strict constraint of narrow sensors is also found not to pose difficulty for real applications of shadow resistant tracking. We will show that such a "rough" approximation is usually good enough to greatly attenuate shadows in tracking an object. A discussion of the model, compared to real cameras and illuminants, is given in Ref. [7].

Invariant image. Consider a Lambertian surface illuminated by a Planckian lighting. The spectral power distribution of a Planckian lighting source can be well approximated by Wien’s approximation [13]:

$$E(\lambda) = I c_1 \lambda^{-5} e^{-c_2/\lambda T}, \quad (1)$$

where I is the intensity of the lighting source, T is temperature, and c_1 and c_2 are constants. We assume that illumination is from direction \mathbf{a} , and that the sensors of the image-capturing device are narrow-band, such that they can be well approximated with spike sensitivities $Q_k(\lambda) = q_k \delta(\lambda - \lambda_k)$, $k = 1, 2, 3$. Here, we consider only the usual 3-sensor cameras; however, our results can be easily extended to the situation where there are more sensors. Further, we ignore the color response change of a point with respect to the viewing angle (i.e., we adopt the Lambertian assumption). Based on the above, the sensor response at point \mathbf{x} corresponding to sensor k is

$$\begin{aligned} \rho_k(\mathbf{x}) &= \int E(\lambda) \mathbf{a} \cdot \mathbf{n}(\mathbf{x}) S_{\mathbf{x}}(\lambda) Q_k(\lambda) d\lambda \quad k = 1..3, \\ &= c_1 \mathbf{a} \cdot \mathbf{n}(\mathbf{x}) I S_{\mathbf{x}}(\lambda_k) \lambda_k^{-5} e^{-c_2/\lambda_k T} q_k, \end{aligned} \quad (2)$$

where $\mathbf{n}(\mathbf{x})$ is the normal vector of the object surface parametrized by image plane coordinates \mathbf{x} , corresponding to surface spectral reflectance function $S_{\mathbf{x}}(\lambda)$. For 3-sensor cameras, it was noted that plotting the set of log-ratios $r = \log[\rho_1/\rho_2]$ and $b = \log[\rho_3/\rho_2]$ for a single surface under various illuminants produces a line in the resulting 2-space [7–9]. Here we explicitly derive this linear relationship between the two log-ratios for a given reflectance surface:

$$\begin{aligned} b - \log \left(\frac{q_3 S(\lambda_3) \lambda_2^5}{q_2 S(\lambda_2) \lambda_3^5} \right) \\ = \left[r - \log \left(\frac{q_1 S(\lambda_1) \lambda_2^5}{q_2 S(\lambda_2) \lambda_1^5} \right) \right] \frac{\lambda_3 \lambda_1 - \lambda_2 \lambda_1}{\lambda_1 \lambda_3 - \lambda_2 \lambda_3}. \end{aligned} \quad (3)$$

From Eq. (3), each reflectance surface is characterized by a line in the coordinate system $(\log(R/G), \log(B/G))$. The slope of the line is determined only by the center wavelengths. In the rest of the paper, we call the orientation of this line, determined by the camera sensors, the *characteristic orientation* for the camera. For a given camera, since the parameters q_k and λ_k , $k = 1..3$ are fixed, the position of the line is determined only by the surface, characterized by the surface spectral reflectance function $S(\lambda)$. Since, for a given camera, all the log–log ratio lines for various surfaces have the same orientation, to distinguish between different materials we simply need to project the log–log ratio points $(\log(R/G), \log(B/G))$ onto the orientation orthogonal to the camera characteristic orientation to produce a quantity encapsulating the different reflectance properties, invariant to the illumination conditions. And if we are invariant to lighting, then we should also be approximately invariant to shadowing.

We call the gray-scale image resulting from transforming the log–log ratio corresponding to each pixel in a color image in this manner the *invariant image*. In the following paragraphs we consider only 3-sensor cameras, and assume $\rho_1 = R$, $\rho_2 = G$, $\rho_3 = B$ and $r = \log[R/G]$, $b = \log[B/G]$.

2.1. Camera calibration

Because different cameras have different color characteristics, they must be calibrated before being used in our tracking scheme. Camera calibration can be done on-line using real videos. However, in many applications, pre-calibration of the camera based on a calibration object is applicable and can give more precise calibration result. In this section, we study camera calibration based on a calibration object for illumination-invariant image generation. For real cameras, the sensor sensitivity curves of sensors are not strictly delta functions and Eq. (3) is thus an approximation. In practical situations, the log–log ratio plot corresponding to one material under different lighting is not strictly a straight line. Nevertheless, amongst a set of materials the log–log ratio plots generally have some fixed dominant orientation. An alternative to camera calibration is to use an entropy minimization technique [14] based on the information in each frame.

From Eq. (3), the dominant orientation is only a function of the camera sensors and does not correlate with the lighting condition or the surface material. Therefore, the calibration process for a specific camera need only be done once and can then be used for any conditions. Here, we omit the effects of gamma correction on images, but in fact this correction has no effect on the mathematics, and simply changes the characteristic orientation [9]. If lines given by Eq. (3) are in practice not quite straight, sensors can be “sharpened” to make them straight [15]. To calibrate, we use a color target comprised of paint patches. Assume \mathcal{P} is the collection of log–log ratio pairs $\{(r_n^i, b_n^i) | n \in \mathcal{N}, i \in \mathcal{I}\}$ where $r_n^i = \log(R_n^i/G_n^i)$ and $b_n^i = \log(B_n^i/G_n^i)$. Here, (R_n^i, G_n^i, B_n^i) is the color of pixel n (or, more realistically, the median of paint patch n) under illumination i in the RGB color space; \mathcal{N} and \mathcal{I} are the whole set of pixel indexes and illumination indexes, respectively. For example, if use the Macbeth ColorChecker target, with 24 patches, then if we image the target under 10 different illumination conditions, $|\mathcal{N}|$ is 24 and $|\mathcal{I}|$ is 10.

We plot curves for patch n under the whole set of illuminants \mathcal{I} . For each patch, we first shift the set of log–log ratio vectors such that the center of the cluster corresponding to one patch under different illuminations is located at the origin of the coordinate system. The centrally aligned log–log ratio set is denoted $\{(\hat{r}_n^i, \hat{b}_n^i) | n \in \mathcal{N}, i \in \mathcal{I}\}$, where

$$\hat{r}_n^i = r_n^i - \frac{1}{|\mathcal{I}|} \sum_{i \in \mathcal{I}} r_n^i, \quad \hat{b}_n^i = b_n^i - \frac{1}{|\mathcal{I}|} \sum_{i \in \mathcal{I}} b_n^i. \quad (4)$$

From now on we assume shifted log–log values, and for conciseness we denote the set by $\{(r, b)\}$. The cross-correlation matrix C of the center-shifted log–log ratio pair set equals

$$C = \begin{bmatrix} e_{rr} & e_{rb} \\ e_{br} & e_{bb} \end{bmatrix}, \tag{5}$$

where $e_{rr} = E[r^2]$, $e_{bb} = E[b^2]$, $e_{rb} = e_{br} = E[rb]$, with $E[\cdot]$ being expectation value. Let μ_1 and μ_2 be eigenvalues of C ; these are as follows:

$$\mu_1 = \frac{1}{2}e_{rr} + \frac{1}{2}e_{bb} + \frac{1}{2}\sqrt{(e_{rr}^2 - 2e_{rr}e_{bb} + e_{bb}^2 + 4e_{rb}^2)},$$

$$\mu_2 = \frac{1}{2}e_{rr} + \frac{1}{2}e_{bb} - \frac{1}{2}\sqrt{(e_{rr}^2 - 2e_{rr}e_{bb} + e_{bb}^2 + 4e_{rb}^2)}.$$

Then the eigenvector v_1 corresponding to μ_1 is defined as the principal orientation of the camera:

$$v = \left(\begin{array}{c} -\frac{1}{2}e_{rr} + \frac{1}{2}e_{bb} - \frac{1}{2}\sqrt{(e_{rr}^2 - 2e_{rr}e_{bb} + e_{bb}^2 + 4e_{rb}^2)} \\ e_{rb} \end{array}, 1 \right)^T.$$

Its orthogonal vector is

$$v_{\perp} = \left(\begin{array}{c} -\frac{1}{2}e_{rr} + \frac{1}{2}e_{bb} + \frac{1}{2}\sqrt{(e_{rr}^2 - 2e_{rr}e_{bb} + e_{bb}^2 + 4e_{rb}^2)} \\ e_{rb} \end{array}, 1 \right)^T.$$

The invariant image is calculated as the gray-scale image projected into the direction v_{\perp} orthogonal to the characteristic direction:

$$I_{inv} = \frac{(\log[R/G], \log[B/G])}{\|v_{\perp}\|} v_{\perp}. \tag{6}$$

3. An inertia snake model

The traditional 2D snake is a deformable curve $X(s) = [x(s), y(s)]$, where s is a parameter in the range $[0, 1]$. The contour is determined by an energy minimization problem. Contour X is that which minimizes the system energy E , defined as

$$E(X) = \int_0^1 \frac{\alpha}{2} \|\nabla X(s)\|^2 + \frac{\beta}{2} \|\nabla^2 X(s)\|^2 + P(X(s)) ds, \tag{7}$$

where α and β are parameters to control the internal tension (stretching) and stiffness (bending) of the contour, respectively; $\nabla X(s) = (dx(s)/ds, dy(s)/ds)$ and $\nabla^2 X(s) = (d^2x(s)/ds^2, d^2y(s)/ds^2)$; $\|(x, y)\| = \sqrt{x^2 + y^2}$; and $P(X)$ is an external energy term which is minimized at the feature of interest. Based on the calculus of variations, the Euler–Lagrange equation corresponding to the variational problem is

$$-\alpha X_{ss} + \beta X_{ssss} + \nabla P(X) = 0. \tag{8}$$

The Euler–Lagrange equation can be solved by the steepest descent method. By introducing an artificial time t , we set

$$\frac{\partial X(s, t)}{\partial t} = \alpha X_{ss} - \beta X_{ssss} - \nabla P(X). \tag{9}$$

The introduction of the time t makes $X(s, t)$ an evolving contour, also called an active contour. The first two terms in the equation are usually called *internal* forces since these forces on the snake are due to a self-shape constraint. The third term is called the *external* force, and is caused by the external features of interest in the image. The external force will attract the snake to converge to a shape near the feature of interest, e.g. edges, while the internal force makes the snake behave like a spline such that it will not spread out and also has some stiffness such that it is not easy to bend. The stiffness term is important for fitting a boundary with holes. But the stiffness also prevents the snake from converging into some concave parts of the boundary of the object. Different snake models are usually distinguished by the scheme used for different external forces [16], and schemes which make the snake able to follow concavities in the object boundary [17].

3.1. Snake equation with predictive contour inertia and chordal string shape descriptor constraint

3.1.1. Predictive contour constraint

In this section, we present the first component of our new model for enhancing the robustness of active contours [18] in the tracking problem. We include a new inertial term in an active contour minimization problem, and then in Section 3.1.2 go on to include an overall shape constraint, based on chords across the object boundary curve. We formulate a new snake minimization problem with an inertial constraint as follows:

$$\min_{X(s)} \int_0^1 \frac{\alpha}{2} \|\nabla X(s)\|^2 + \frac{\beta}{2} \|\nabla^2 X(s)\|^2 + P(X(s)) + \frac{\gamma}{2} E(X(s), C(s)) ds, \tag{10}$$

where $X(s) = [x(s), y(s)]$ is the active contour of the current frame, and $C(s) = [c_1(s), c_2(s)]$ is the prediction contour from the previous frame. $E(X(s), C(s))$ is a term which measures the difference between $X(s)$ and $C(s)$ [19]. Just as in a traditional snake, the internal energy of the active contour is handled by the first two terms, with weights α and β . The term $P(X(s))$ is the external force, based on a feature of interest such as edge information; here we shall use object motion, instead, as the driving feature.

One natural choice for $E(X(s), C(s))$ is the distance

$$E(X(s), C(s)) = \|X(s) - C(s)\|^2. \tag{11}$$

If the norm chosen is the Euclidean norm $\|(x, y)\| = \sqrt{x^2 + y^2}$, the corresponding Euler–Lagrange equation is

$$-\alpha X_{ss} + \beta X_{ssss} + \nabla P(X) - \gamma(C - X) = 0 \tag{12}$$

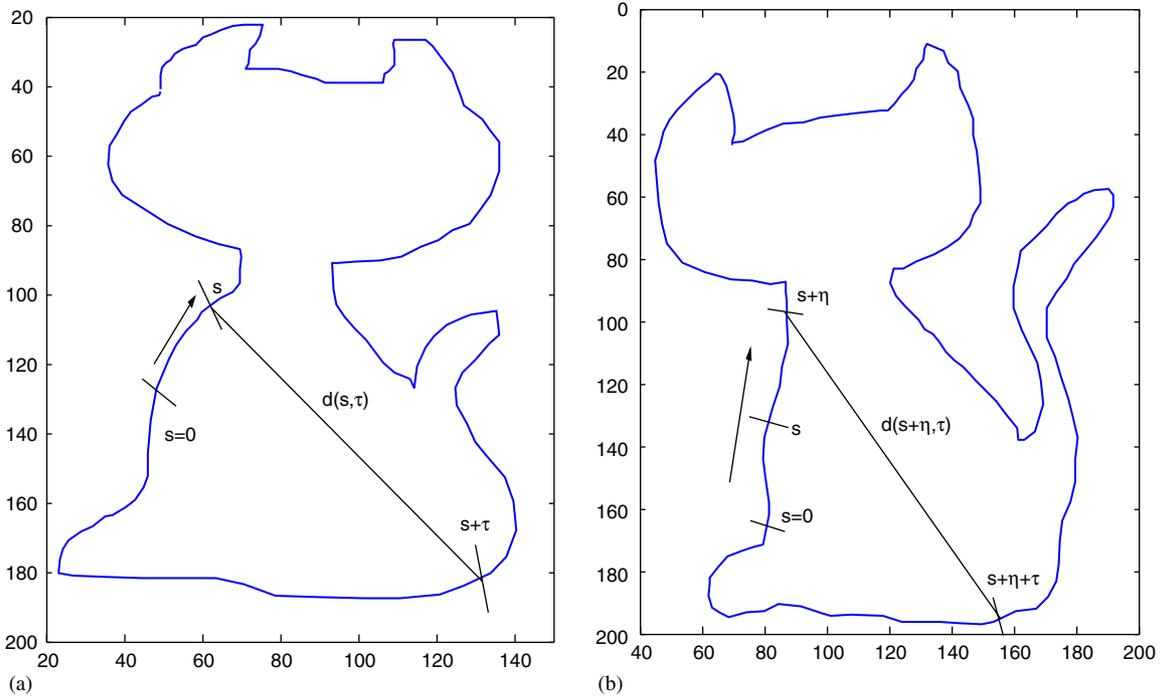


Fig. 1. Chordal distance. Chord $d(s, \tau)$ is a function of position and shift.

and by introducing an artificial parameter t into X , the steepest descent solution is

$$\frac{\partial X}{\partial t} = \alpha X_{ss} - \beta X_{ssss} - \nabla P(X) + \gamma(C - X). \quad (13)$$

We generalize the above equation by substituting a general force term $F_{ext}(X)$ in place of the potential term, $-\nabla P(X)$, leading to the final equation

$$\frac{\partial X}{\partial t} = \alpha X_{ss} - \beta X_{ssss} + F_{ext}(X) + \gamma(C - X). \quad (14)$$

This yields a modified active contour for the tracking problem, with a new force term based on the prediction contour. Weight γ controls the influence the prediction has on contour tracking.

3.1.2. Chordal string shape descriptor constraint

In this section we further constrain the shape compactness of the snake and its shape persistence from frame to frame, by the use of a new chordal string shape descriptor constraint. To do so, we iterate through a contour evolution that tries to maintain a *chordal string descriptor*.

In practical situations, the contour of an object does not change arbitrarily: the shape of the contour usually displays some degree of rigidity. As we have seen, traditional snakes control the rigidity of the snake using two terms, the stiffness and the smoothness, by means of terms in the gradient and the Laplacian of the contour. Since the gradient and Laplacian are *local* operators, the *global* shape of the snake is thus usually not controlled effectively. Here, we propose

a scheme based on constraining the shape of the contour by means of a new set of shape descriptors based on chordal strings. We first present a new shape descriptor and then present a new intra-/inter-frame constraint snake equation.

Chordal string shape descriptor. Let us define a new shape descriptor by considering the following observations. Firstly, a closed contour $X(s)$ is defined as a periodic function in this context, with s the normalized arc length along the contour starting from some arbitrarily selected origin. Then the following function:

$$d(s, \tau) = \|X(s) - X(s + \tau)\| \quad (15)$$

is also periodic, with respect to both s and τ , with a period of unity. This *chordal distance* $d(s, \tau)$ is a natural choice for characterizing the shape of a closed contour. The chordal distance $d(s, \tau)$ for a particular s and τ is shown in Fig. 1(a).

Then the difference of two contours X and Y can be defined as the sum of the square differences in chordal distances, over all shifts τ , along each complete curve:

$$\mathcal{D}(X, Y) = \min_{\eta \in (0,1)} \int_0^1 \int_0^1 (d_X(s, \tau) - d_Y(s + \eta, \tau))^2 ds d\tau. \quad (16)$$

Parameter η is a shift necessary because of the arbitrarily selected origin. This measure is shown in Fig. 1.

Now we have a shape descriptor for general shape comparison. In the following scheme, we use a restricted version of the proposed shape descriptor so as to make the scheme suitable for fast video processing. In the rest of this section,

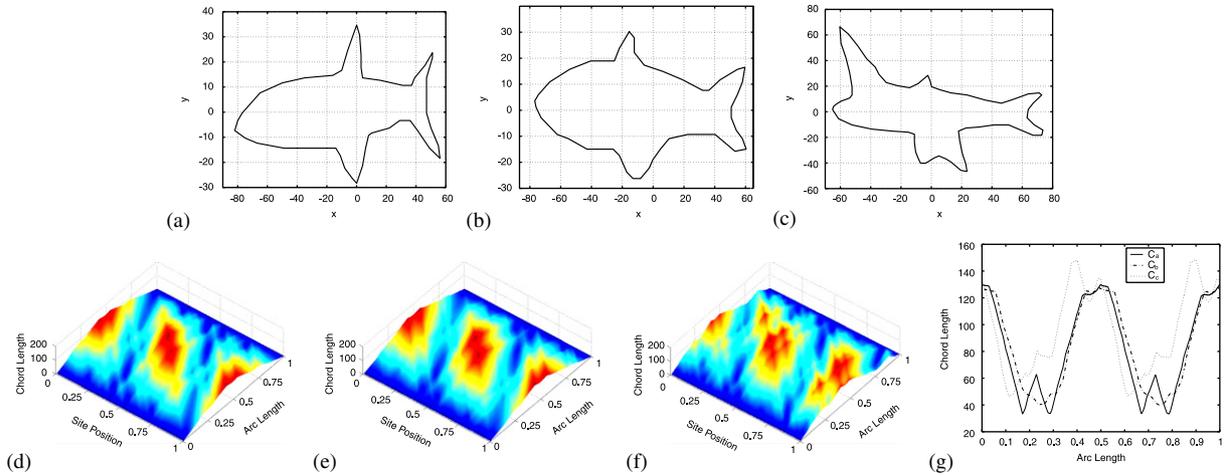


Fig. 2. Chordal string shape descriptor. (a) a fish contour; (b) a fish contour similar to (a); (c) a contour dissimilar to (a); (d) chordal string shape descriptor for contour (a); (e) chordal string shape descriptor for contour (b); (f) chordal string shape descriptor for contour (c); and (g) simplified chordal shape descriptor with $\tau = 1/2$ for contours (a), (b) and (c).

we fix τ as $1/2$. η can also be set to 0 if we know the corresponding starting point of the contours. To simplify the notation, we use $d(s)$ to denote $d(s, 1/2)$. The intuitive concept of $d(s)$ is that it defines the “diameter” of the contour at different locations. For a circular contour, $d(s)$ is just a constant function d , the diameter of the circle. Then the difference of contour X and Y can be represented as

$$\mathcal{D}(X, Y) = \int_0^1 (d_X(s) - d_Y(s))^2 ds. \tag{17}$$

Fig. 2 illustrates an example of measuring shape with the proposed shape descriptor. Figs. 2 (a)–(c) show three shapes; while the contour of (a) and (b) are similar, (a) and (c) are more different. The proposed shape descriptors, shown in Figs. 2 (d)–(f) agree with this observation. The shape differences based on the proposed area-based shape descriptor (16) are 6.07 for contours (a) and (b), and 17.44 for contours (a) and (c). The simplified line-based shape descriptors (17) with τ fixed to be 0.5 are shown as curves in Fig. 2 (g). For the simplified descriptor, the shape difference is 7.66 for contours (a) and (b) and 23.53 for contours (a) and (c).

The snake equation. Suppose we have determined a predictive contour $C(s)$. Then we have still to find the best snake solution $X(s)$ in the current frame that takes into account our inter-frame constraints: i.e., we need a steady-state solution that yields the best curve path given the current, intra-frame, information.

Recall that our objective in using the chordal string descriptor as defined above is to maintain the shape of contour across frame change. We have found that the addition of such a global shape constraint mechanism *substantially increases tracking reliability*. Section 3.4 explicates how this extra constraint operates.

If we specialize to “diameters”, as above, then we claim that the equation we wish to solve, using curve evolution, for a correct curve within the current frame is that derived using the following proposition.

Proposition. *The partial differential equation for curve $X(s)$ that includes diameter chord shape descriptor constraints is as follows:*

$$\begin{aligned} \frac{\partial X}{\partial t} = & \alpha X_{ss} - \beta X_{ssss} + F_{ext}(X) + \gamma(C - X) \\ & + \rho \frac{(X(s + 1/2) - X(s))(\|X(s) - X(s + 1/2)\| - d_{X-1}(s))}{\|X(s + 1/2) - X(s)\| + \varepsilon}, \end{aligned} \tag{18}$$

where ε is a small positive number.

Proof. To prove this result, first let us utilize a fictitious auxiliary curve $Y(s)$, one that we wish to equal a copy of $X(s)$, but shifted by half of the contour length: i.e., we wish to have $X(s) = Y(s + 1/2)$. First, let us start with a curve $Y(s)$ that is free, but initialized as a curve $Y_0(s)$ given by the diameter-shifted version of the initial curve $X_0(s)$ for $X(s)$ itself: $Y_0(s) \equiv X_0(s + 1/2)$.

Then a reasonable new snake equation is represented as the following reaction-diffusion problem:

$$\begin{aligned} \min_{X(s), Y(s)} \int_0^1 & \frac{\alpha}{2} (\|\nabla X(s)\|^2 + \|\nabla Y(s)\|^2) \\ & + \frac{\beta}{2} (\|\nabla^2 X(s)\|^2 + \|\nabla^2 Y(s)\|^2) + P(X(s)) \\ & + P(Y(s)) + \frac{\gamma}{2} \{E(X(s), C(s)) + E(Y(s), D(s))\} ds \\ & + \frac{\rho}{2} \int_0^1 G(\|X(s) - Y(s)\|, d_{X-1}(s)) ds, \end{aligned}$$

where $X(s)$ is the active contour in the current frame; $Y(s)$ is an auxiliary contour coupled with $X(s)$; and $C(s)$ and $D(s)$ are the prediction contours for X and Y from the previous frame. Functions $E(\cdot, \cdot)$ and $G(\cdot, \cdot)$ are two energy terms. Energy $E(\cdot, \cdot)$ was defined in the last section. The term $G(\cdot, \cdot)$ is defined as

$$G(A(s), B(s)) = \|A(s) - B(s)\|^2$$

with $d_{X^{-1}}(s, \tau)$ our distance function (15) and $d_{X^{-1}}(s)$ evaluated using the *previous* frame's $X(s)$. We will show that $\|X(s) - Y(s)\| = d_X(s)$, and therefore $\int_0^1 G(\|X(s) - Y(s)\|, d_{X^{-1}}(s)) ds = \mathcal{D}(X, X^{-1}) = \mathcal{D}(Y, Y^{-1})$, where X^{-1} and Y^{-1} are the contours of X and Y in the previous frame, respectively. That is, we wish to maintain the chordal shape descriptor, as the contour evolves, thus encouraging the maintenance of a global shape.

In a Euclidean norm, the resulting set of Euler equations is

$$\begin{aligned} & -\alpha X_{ss} + \beta X_{ssss} + \nabla P(X) - \gamma(C - X) \\ & - \rho \frac{(Y(s) - X(s))}{\|X(s) - Y(s)\|} (\|X(s) - Y(s)\| - d_{X^{-1}}(s)) = 0, \\ & -\alpha Y_{ss} + \beta Y_{ssss} + \nabla P(Y) - \gamma(D - Y) \\ & - \rho \frac{(X(s) - Y(s))}{\|X(s) - Y(s)\|} (\|X(s) - Y(s)\| - d_{X^{-1}}(s)) = 0. \end{aligned}$$

As usual, we replace $-\nabla P(X)$ in the above equations by a generalized force term $F_{ext}(X)$. Curve evolution is modeled using a fictitious time variable t , with change in time set to the left-hand sides above. Thus, the resulting iterative steepest descent solution is as follows:

$$\begin{aligned} \frac{\partial X}{\partial t} &= \alpha X_{ss} - \beta X_{ssss} + F_{ext}(X) + \gamma(C - X) \\ &+ \rho \frac{(Y - X)}{\|X - Y\|} (\|X - Y\| - d_{X^{-1}}), \\ \frac{\partial Y}{\partial t} &= \alpha Y_{ss} - \beta Y_{ssss} + F_{ext}(Y) + \gamma(D - Y) \\ &+ \rho \frac{(X - Y)}{\|X - Y\|} (\|X - Y\| - d_{X^{-1}}). \end{aligned} \quad (19)$$

Suppose the initial state of $X(s)$ is $X_0(s)$; then let the initial state of $Y(s)$ be $X_0(s + 1/2)$. Similarly, let the initial state of D be $D(s) = C(s + 1/2)$. The shape descriptor $d_{X^{-1}}(s)$ is that for the previous frame. Note that the solution $X(s)$ is the tracking contour, while $Y(s)$ is an ancillary contour. We can now state an observation allowing us to dispense with $Y(s)$.

Observation 1. *If the initial values of X and Y obey the condition $X(s, t_0) \equiv Y(s + 1/2, t_0)$, then we also have $X(s, t) = Y(s + 1/2, t)$ for any $t > t_0$.*

Proof. We will prove the proposition by induction. Assume, for any $t_1 \leq t$, that $X(s, t_1) = Y(s + 1/2, t_1)$. Then,

$$\begin{aligned} X(s, t_1 + dt) &= [\alpha X_{ss}(s, t_1) - \beta X_{ssss}(s, t_1) \\ &+ F_{ext}(X(s, t_1)) + \gamma(C(s, t_1) - X(s, t_1)) \\ &+ \rho \frac{(Y - X)}{\|X - Y\|} (\|X - Y\| - d)] dt + X(s, t_1) \end{aligned}$$

and

$$\begin{aligned} Y(s, t_1 + dt) &= [\alpha Y_{ss}(s, t_1) - \beta Y_{ssss}(s, t_1) \\ &+ F_{ext}(Y(s, t_1)) + \gamma(D(s, t_1) - Y(s, t_1)) \\ &+ \rho \frac{(X - Y)}{\|X - Y\|} (\|X - Y\| - d)] dt + Y(s, t_1). \end{aligned}$$

Making use of the periodic property of X and Y , we have $X(s + 1/2, t_1) = Y((s + 1/2) + 1/2, t_1) = Y(s, t_1)$. At a slightly later time, it is also easy to verify that $X(s, t_1 + dt) = (s + 1/2, t_1 + dt)$. By induction, the lemma follows. \square

Observation 2. *The coupled set of Eq. (19) has the same solution X as the PDE (18).*

Proof. Eq. (18) follows from the lemma above, with no constant ε . That constant is needed, however, as a regularization parameter: adding a small positive number ε to the denominator prevents overflow if $\|X(s) - Y(s)\|$ is near zero. \square

This completes the proof of the proposition, and shows that we can indeed control the shape of the contour according to the defined shape descriptor. We do not need to solve for both of X and Y , but instead the coupled set of Eq. (19) can be simplified into a single PDE. Thus, Eq. (18) is the equation we utilize in the contour tracking procedure. The numerical method used for solving the PDE problem is discussed below in Section 3.5.

The equations together try to maintain the shape of the contour from frame to frame via our new chordal string constraint. Weight ρ controls the degree of shape rigidity. This modified active contour for the tracking problem also includes the new inertia force term based on the prediction contour: weight γ controls the degree of influence of the prediction on contour tracking.

3.2. Contour prediction based on ICM

So far, we have simply assumed that we have available a predictive contour, derived from the previous frame. Now let us examine a method for producing such a prediction. Since we have several constraints to maintain the shape of the snake, we found that the precision of the predictive contour could be relaxed. In this paper, we use a fast contour matching method based on ICM [20].

The contour prediction problem can be formulated as the following energy minimization problem,

$$\min_{\mathbf{v}} \sum_{\mathbf{x} \in \mathcal{X}} e(\mathbf{x}, \mathbf{v}) + \sum_{(\mathbf{x}, \mathbf{y}) \in \mathcal{N}} \lambda(\mathbf{x}, \mathbf{y}) \|\mathbf{v}(\mathbf{x}) - \mathbf{v}(\mathbf{y})\|^2,$$

where X is the contour in the current frame; $e(\mathbf{x}, \mathbf{v})$ is the cost of site \mathbf{x} having motion \mathbf{v} , and defined as the block matching result in a rectangular searching window; \mathcal{N} is the set of neighbors; $\lambda(\mathbf{x}, \mathbf{y})$ is a smoothing coefficient which decreases as the distance of \mathbf{x} and \mathbf{y} increases, e.g. $\lambda(\mathbf{x}, \mathbf{y}) = \lambda_0 e^{-\|\mathbf{x}-\mathbf{y}\|/\tau}$, where τ determines the coupling strength of the sites. It should be noted that \mathcal{N} need not be defined as strictly adjacent sites on the contour: in this paper, \mathcal{N} is defined by the edges of the Delaunay triangulation of the sites on the contour. Neighbors are nodes on the contour within this triangulation. For a general topology, the optimization problem is *NP*-hard. We use the ICM scheme to get an approximate solution. The initial values of \mathbf{v} are set to the best local block searching result. The typical width of the block is 5–11. For each site, all possible motions for the site are tested and the one that most reduces the energy is accepted. Since there is no guarantee that iteration will converge for ICM, a heuristic large iteration number should be chosen. In our experiments, we found that 20 iterations suffice. Based on the calculated motion, the predicted contour is

$$C = X^{-1} + \mathbf{v}. \quad (20)$$

We modify the initial contour C_{init} to be equal to a uniform expansion of the previous contour tracking result:

$$C_{init} = X^{-1} + c\mathbf{n}, \quad (21)$$

where \mathbf{n} is the outward-directed normal of X_{prev} , and c is a constant. It is clear that this scheme can work for any motion of the contour with speed less than $c \cdot f$ in the normal direction, where f is the video frame rate.

3.3. Global motion compensation and external force formation

To remove the motion introduced by the camera, called global motion, we need to estimate the motion in the whole picture. If the background object is a plane, then the background image in one frame can be mapped to another frame by a projective transformation or a homography. Although the projective transformation is only exact for a plane, it is often used to approximate the background motion if the object is distant from the camera. But the projective transformation is difficult to estimate since the denominator makes for a nonlinear transformation. In most circumstances, if the object is far away relative to the focal length of the camera and the viewing angle is small so that depth does not change too much, an affine model is a reasonable approximation. In this case we have a 6-parameter affine model. The affine flow field can be represented as

$$\theta(x, y) = A(x, y)\mathbf{p}, \quad (22)$$

where $\mathbf{p} = \{p_1, p_2, p_3, p_4, p_5, p_6\}^T$ are the parameters to be estimated. Here,

$$A(x, y) = \begin{pmatrix} 1 & x & y & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & x & y \end{pmatrix}. \quad (23)$$

The optical flow equation can be written as

$$\nabla u^T(A\mathbf{p}) + u_t = 0. \quad (24)$$

This is an over-determined problem, in that we have more equations than the number of unknowns. We can use a *least square error* model to get an approximate solution, minimizing an E defined by

$$E = \sum_{x,y} (\nabla u^T(A\mathbf{p}) + u_t)^2. \quad (25)$$

Taking the derivative of E with respect to \mathbf{p} and setting the result to zero, the solution for the affine parameters is

$$\mathbf{p} = \left(\sum A^T \nabla u \nabla u^T A \right)^{-1} \sum (-A^T \nabla u u_t). \quad (26)$$

In our tracking scheme, we use only the part of the image *outside the contour of the object* for global motion estimation. This method is straightforward for our scheme, since we already extract the moving object region in the image when tracking. For global motion estimation, it is enough to downsample the input images and estimate the global motion parameters at a very coarse level. In the proposed tracking scheme, two motion detection maps are generated. The first one is based on the original video with global motion compensated. The second is based on the shadow invariant global motion compensated images. We then use a simple thresholding scheme to detect the motion feature in both sequences. The intersection of both these motion detection results produces an image segmentation map which is used to calculate the external force field of the snake model, based on the gradient vector flow scheme [17].

3.4. Evaluation of chordal constraint

To show that our chordal string constraint does indeed increase tracking reliability, we test the method with and without this new constraint.

Consider the U-shaped object in Fig. 3(a). If we take as initial curve the green line shown in Fig. 3(b),¹ then the version that is expanded in the normal direction according to the contour initialization method (21) is shown in blue, expanded outside the figure. We assume that there is an error of prediction contour occurring near the concave part of the object. And, we assume the object is static. Now applying the active contour model Eq. (14), with the contour prediction inertia term $(C - X)$ but without chordal strings, yields the thicker, red, curve in Fig. 3(c). This result includes calculating the external force field of the snake model based on the gradient vector flow scheme, which is indeed meant to help in attracting the curve down into such deep concavities.

¹ Figures in color are included in <http://www.cs.sfu.ca/cs/~mark/ftp/PR05/>

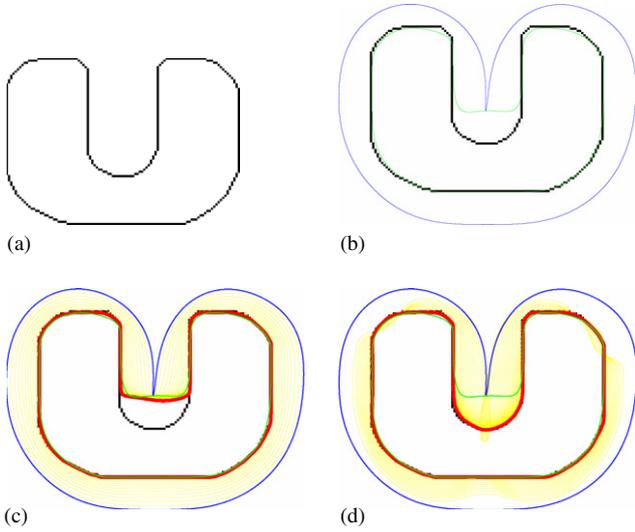


Fig. 3. Illustration of efficacy of chordal constraints. (a) target object; (b) contour initialization—green; initialization expanded according to (21)—blue; (c) with inertial constraint: red—solution of curve evolution (14); and (d) with chordal string constraints as well. The yellow curving are evolving contours.

We see that even with this help, the active contour scheme has not completely succeeded.

On the other hand, including chordal string constraints instead generates the result in Fig. 3(d): now the solution curve almost coincides with the objective shape.

$$A = \begin{bmatrix} -\frac{2\alpha\Delta t}{\Delta s^2} - \frac{6\beta\Delta t}{\Delta s^4} & \frac{\alpha\Delta t}{\Delta s^2} + \frac{4\beta\Delta t}{\Delta s^4} & -\frac{\Delta t\beta}{\Delta s^4} & -\frac{\beta\Delta t}{\Delta s^4} & \frac{\alpha\Delta t}{\Delta s^2} + \frac{4\beta\Delta t}{\Delta s^4} \\ \frac{\alpha\Delta t}{\Delta s^2} + \frac{4\beta\Delta t}{\Delta s^4} & -\frac{2\alpha\Delta t}{\Delta s^2} - \frac{6\beta\Delta t}{\Delta s^4} & \frac{\alpha\Delta t}{\Delta s^2} + \frac{4\beta\Delta t}{\Delta s^4} & -\frac{\Delta t\beta}{\Delta s^4} & -\frac{\beta\Delta t}{\Delta s^4} \\ \frac{\alpha\Delta t}{\Delta s^2} + \frac{4\beta\Delta t}{\Delta s^4} & -\frac{\Delta t\beta}{\Delta s^4} & \dots & -\frac{\beta\Delta t}{\Delta s^4} & \frac{\alpha\Delta t}{\Delta s^2} + \frac{4\beta\Delta t}{\Delta s^4} \\ \dots & \dots & \dots & \dots & \dots \\ \frac{\alpha\Delta t}{\Delta s^2} + \frac{4\beta\Delta t}{\Delta s^4} & -\frac{\Delta t\beta}{\Delta s^4} & -\frac{\beta\Delta t}{\Delta s^4} & \frac{\alpha\Delta t}{\Delta s^2} + \frac{4\beta\Delta t}{\Delta s^4} & -\frac{2\alpha\Delta t}{\Delta s^2} - \frac{6\beta\Delta t}{\Delta s^4} \end{bmatrix}.$$

3.5. Numerical implementation

We use the finite difference method to discretize Eq. (18). Let

$$X_i^n = (x_i^n, y_i^n) = (x(i\Delta s, n\Delta t), y(i\Delta s, n\Delta t)),$$

$$d_i = d(x_i^n, y_i^n). \tag{27}$$

As usual, we assume that X_i^n is periodic with period of N , with N the number of nodes in the discrete contour. Thus, we have $X_i^n = X_{i \bmod N}^n$. The numerical form for Eq. (14)

is then

$$\begin{aligned} \frac{X_i^{n+1} - X_i^n}{\Delta t} = & \frac{\alpha}{\Delta s^2} [X_{i+1}^{n+1} + X_{i-1}^{n+1} - 2X_i^{n+1}] \\ & - \frac{\beta}{\Delta s^4} \left\{ [X_{i+2}^{n+1} + X_i^{n+1} - 2X_{i+1}^{n+1}] \right. \\ & - 2[X_{i+1}^{n+1} + X_{i-1}^{n+1} - 2X_i^{n+1}] \\ & \left. + [X_i^{n+1} + X_{i-2}^{n+1} - 2X_{i-1}^{n+1}] \right\} \\ & + F_{ext}(X_i^n) + \gamma(C_i^n - X_i^n) \\ & + \rho \frac{(X_{i+N/2}^n - X_i^n)}{\|X_{i+N/2}^n - X_i^n\| + \varepsilon} \\ & \times (\|X_{i+N/2}^n - X_i^n\| - d_i). \end{aligned} \tag{28}$$

Let $x^n = [x_0^n, x_1^n, \dots, x_{N-1}^n]^T$, $y^n = [y_0^n, y_1^n, \dots, y_{N-1}^n]^T$ and $X^n = [x^n, y^n]$, $Y^n = [X_{i+N/2}^n]$ and $T^n = [\|X_{i+N/2}^n - X_i^n\|]$. In matrix form, we have,

$$X^{n+1} - X^n = AX^{n+1} + \Delta t F_{ext}(X^n) + \Delta t \gamma(C^n - X^n) + \rho \Delta t \frac{(Y^n - X^n)}{T^n + \varepsilon} (T^n - d_i). \tag{29}$$

Solving for X^{n+1} ,

$$X^{n+1} = (I - A)^{-1} [X^n + \Delta t F_{ext}(X^n) + \Delta t \gamma(C^n - X^n) + \rho \Delta t \frac{(Y^n - X^n)}{T^n + \varepsilon} (T^n - d_i)], \tag{30}$$

where

$$\text{where } A = \begin{bmatrix} -\frac{2\alpha\Delta t}{\Delta s^2} - \frac{6\beta\Delta t}{\Delta s^4} & \frac{\alpha\Delta t}{\Delta s^2} + \frac{4\beta\Delta t}{\Delta s^4} & -\frac{\Delta t\beta}{\Delta s^4} & -\frac{\beta\Delta t}{\Delta s^4} & \frac{\alpha\Delta t}{\Delta s^2} + \frac{4\beta\Delta t}{\Delta s^4} \\ \frac{\alpha\Delta t}{\Delta s^2} + \frac{4\beta\Delta t}{\Delta s^4} & -\frac{2\alpha\Delta t}{\Delta s^2} - \frac{6\beta\Delta t}{\Delta s^4} & \frac{\alpha\Delta t}{\Delta s^2} + \frac{4\beta\Delta t}{\Delta s^4} & -\frac{\Delta t\beta}{\Delta s^4} & -\frac{\beta\Delta t}{\Delta s^4} \\ \dots & \dots & \dots & \dots & \dots \\ -\frac{\beta\Delta t}{\Delta s^4} & \frac{\alpha\Delta t}{\Delta s^2} + \frac{4\beta\Delta t}{\Delta s^4} & -\frac{2\alpha\Delta t}{\Delta s^2} - \frac{6\beta\Delta t}{\Delta s^4} & \dots & \dots \end{bmatrix}.$$

The parameters such as the $\alpha\Delta t/\Delta s^2$, $\beta\Delta t/\Delta s^4$, $\Delta t\gamma$, Δt and ρ should be determined differently for specific applications. In our numerical scheme the typical values for $\alpha\Delta t/\Delta s^2$, $\beta\Delta t/\Delta s^4$, $\Delta t\gamma$, Δt and $\Delta t\rho$ are 0.4, 0.4, 0.15, 0.6 and 0.5.

4. Shadow resistant tracking system

In this section we present the overall system for our tracking scheme. The system diagram is shown in Fig. 4.

The system is based on an incremental scheme. Two consecutive frames are used in the global motion estimation, motion detection, and contour prediction steps. The scheme can also be easily extended to the three-frame or multi-frame model. The shadow resistant tracking algorithm is as follows:

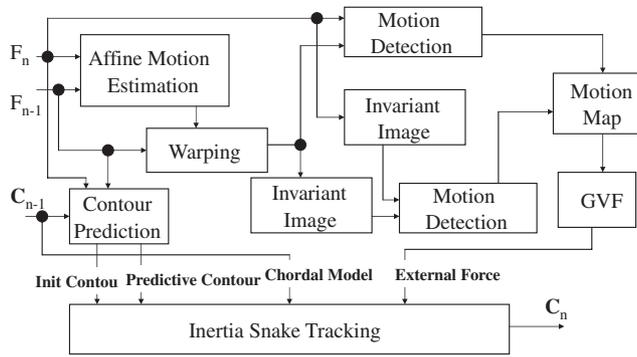


Fig. 4. System diagram.

fixed such that we could retain the pixel correspondence under different illuminations. The analog video was then digitized. We manually segmented the images into regions corresponding to different color blocks. The color in each region was represented by the mean R , G , and B value. Fig. 9 shows that a scatter plot of the center-shifted log–log ratio data gives the illumination invariant orientation of the camcorder. The orientation vector measured was $[0.37, -0.93]$. The center-shifted log–log ratios corresponding to each material are fit fairly well by a straight-line, with some outliers. Errors are caused by several factors. One is the model error. Since the straight line model is exact only for ideal pulse

Algorithm 1.

1. Fetch frame i , denoted by $F(i)$; previous frame is $F(i - 1)$.
2. Calculate the affine transformation from $F(i - 1)$ to $F(i)$, denoted as H , based on Eq. (26).
3. Calculate the first motion map, $map1 = |F(i) - Warp(F(i - 1), H)| > threshold1$.
4. Calculate the second motion map:
 $map2 = |Invariant(F(i)) - Invariant(Warp(F(i - 1), H))| > threshold2$.
5. Combined motion map is: $map = intersection(map1, map2)$.
6. Generate $F_{ext} = GVF(map)$.
7. Contour prediction based on ICM and initiate contour generation (21).
8. Calculate $d(s)$ from $F(i - 1)$.
9. If $F(i)$ is the first frame, manually draw the contour of the object.
 snake equation is $\alpha X_{ss} - \beta X_{sss} + F_{ext}(X) = 0$.
 Else the snake evolves according to Eq. (18).
10. If the procedure does not finish go to 1, else exit.

Values $threshold1$ and $threshold2$ should be determined by experiment. Typical values are 20/255 and 50/255 for normalized gray-scale images.

5. Experimental results

First, we applied the proposed tracking method to video sequences with little shadow interference. Both moving and fixed camera situations are tested. In these experiments, only the source color video is used to estimate object motions, and not the shadow-invariant version. Figs. 5–8 show the tracking results based on the proposed scheme.

For shadow resistant object tracking, we used a consumer camcorder (Canon ES60) in our experiments—the method is robust against gamma correction and can be used in general environments. Before the tracking experiments, we first calibrated the camcorder to obtain the shadow invariant orientation. We used a Macbeth ColorChecker color target with 24 color patches, imaged under three illuminations, for the camera calibration. The images for the calibration are shown in Fig. 5.

The illuminants used were a standard daylight and two different indoor lights. (Alternatively, one could simply capture images as the daylight phase changes, outdoors.) During the video capture process, both the camera and the target were

sensor cameras, the actual wide-band sensors will cause dispersion of the log–log ratios. Other errors come from the measuring limitations posed by the camera’s dynamic range and digitizer quantization. As shown in the following experiments, the estimated orientation is nevertheless robust enough to greatly improve the tracking result.

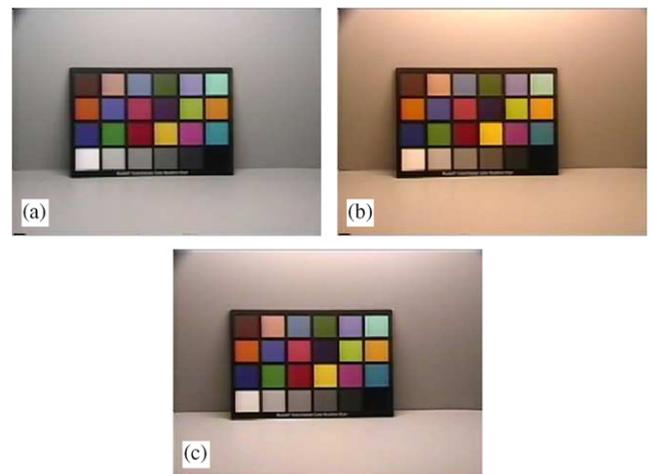


Fig. 5. The color target used for camera calibration. (color images may be viewed at www.cs.sfu.ca/~mark/ftp/PR05/shadowlesstracking05.pdf). (a) Illumination 1; (b) Illumination 2; and (c) Illumination 3.

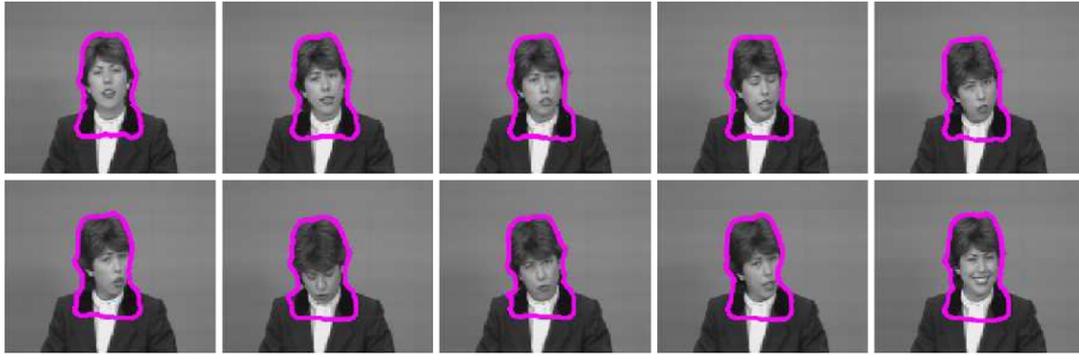


Fig. 6. Selected frames from tracking result for the Claire sequence.

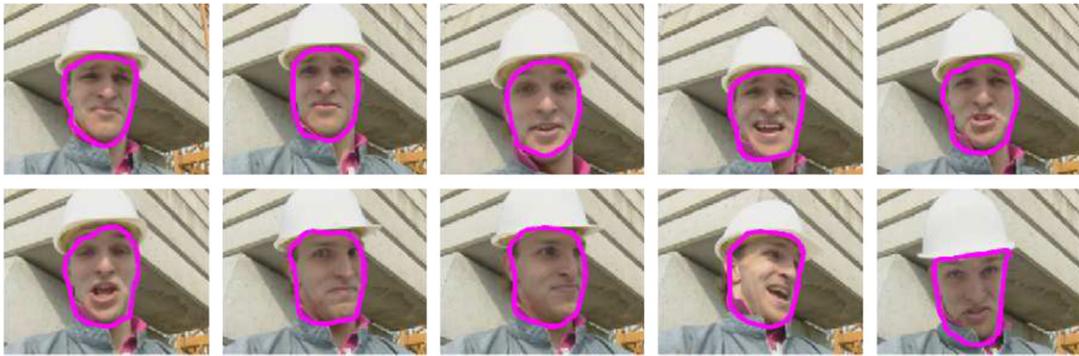


Fig. 7. Selected frames from tracking result for the Foreman sequence.



Fig. 8. Selected frames from tracking result for the Lab sequence. Selected frames from a 470 frame sequence.

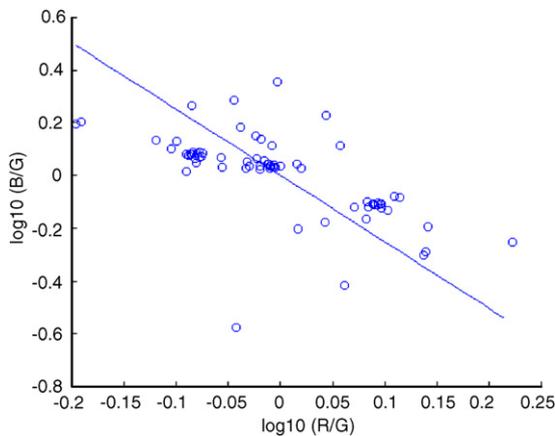


Fig. 9. Camera calibration to find characteristic orientation.

Fig. 10 shows the motion detection result for a two-color ball rolling on the ground, based on the original image sequence and the illumination invariant sequence. The traditional motion detection scheme makes large errors on both the object's boundary and the shadow boundary. Motion detection based on the shadow-invariant image obtains much better results. The shadow's influence is almost completely removed. We also note that the background of the shadow-invariant image motion detection result is also much clearer. To increase the robustness we use the intersection of the motion detection map of the original image sequence and that for the shadow-invariant image sequence as the final motion detection map.

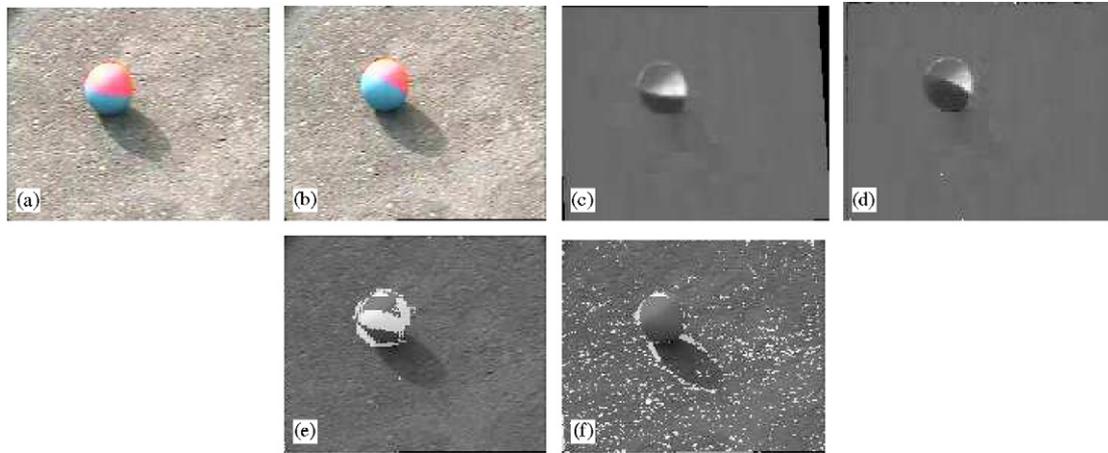


Fig. 10. Motion detection map: (a, b) two consecutive frames; (c) warped shadow-invariant image for frame (a); (d) shadow-invariant image for frame (b); (e) motion map by shadow-invariant image overlapped with frame (b); and (f) motion map by original gray-scale images overlapped with frame (b).

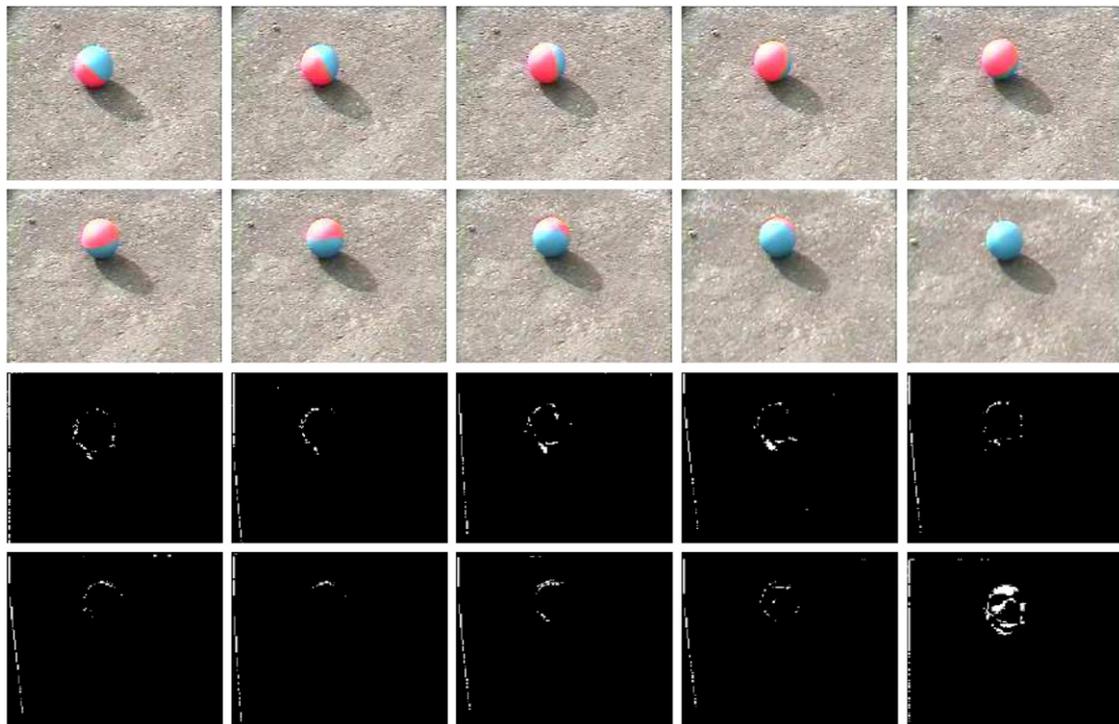


Fig. 11. Ten-frame test sequence and corresponding motion map. Note that the motion features are missing in this example.

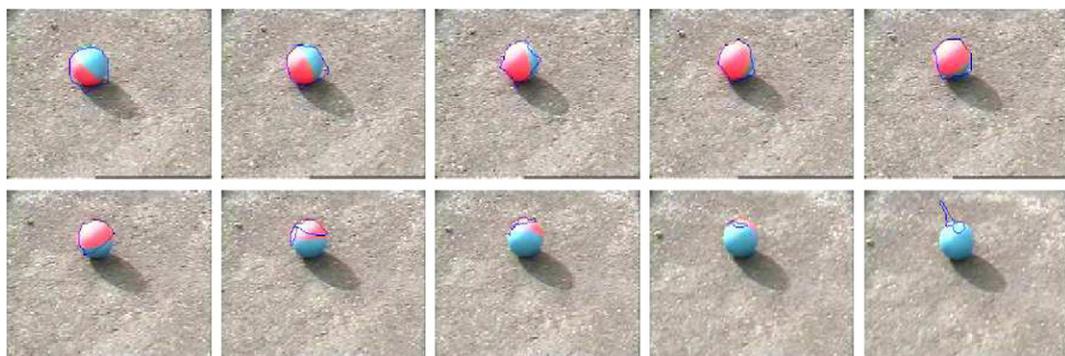


Fig. 12. Tracking result with traditional snake for the ten-frame test sequence.

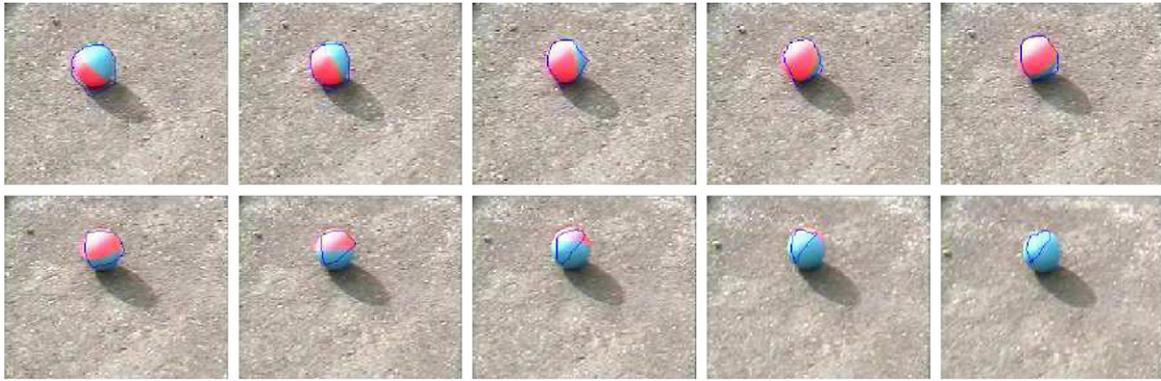


Fig. 13. Tracking result only with the predictive contour inter-frame constraint term for the test sequence.

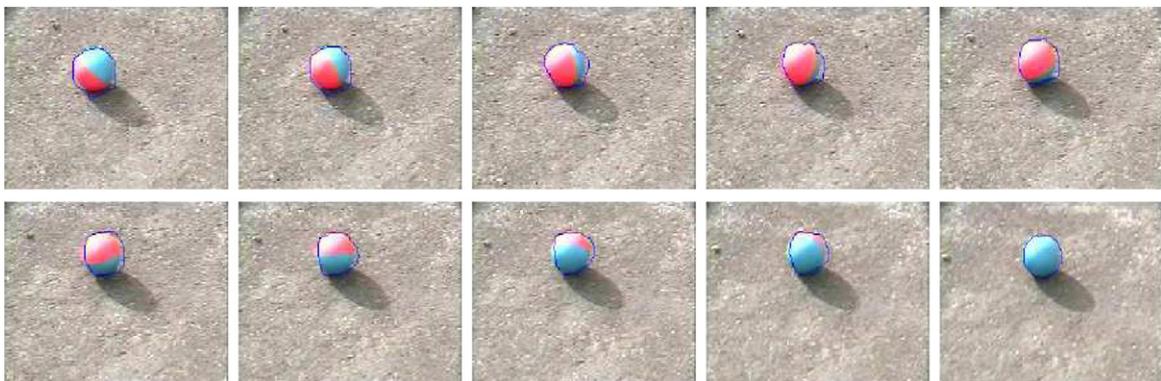


Fig. 14. Tracking result with both the predictive contour inter-frame constraint and chordal shape descriptor inter-frame constraint for the test sequence.

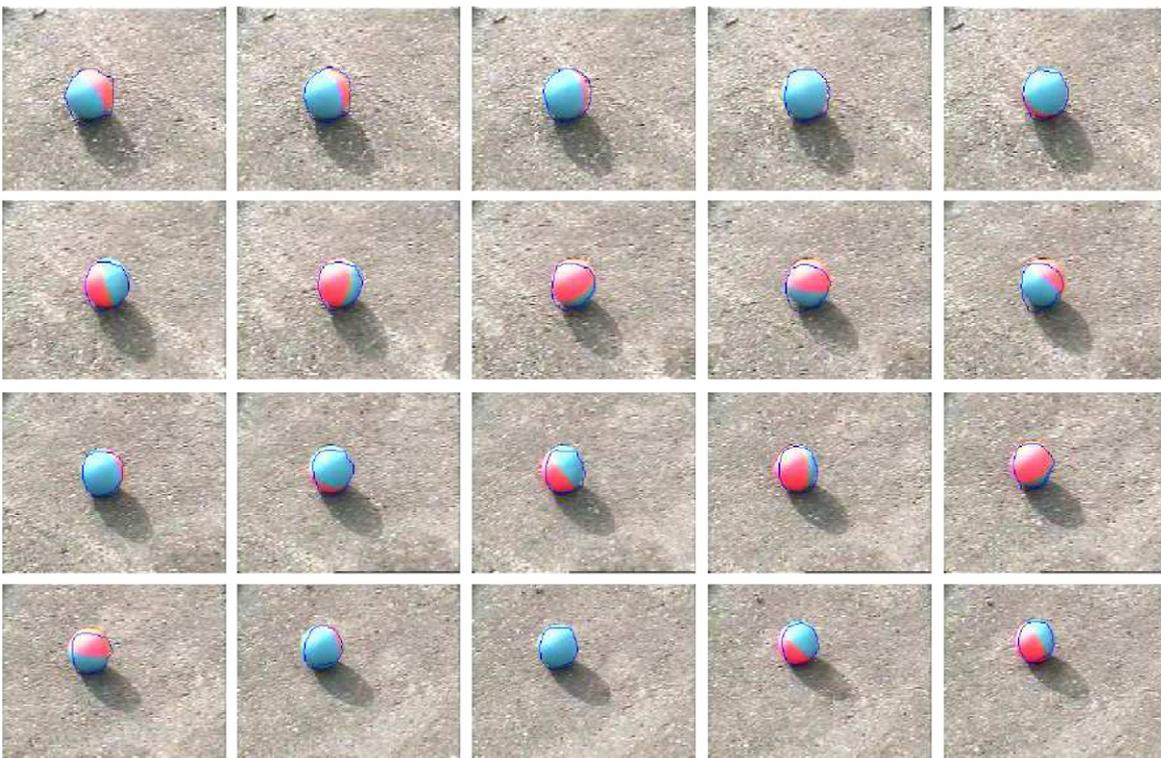


Fig. 15. Tracking result with proposed method for the ball sequence. Selected frames from the 40-frame sequence.

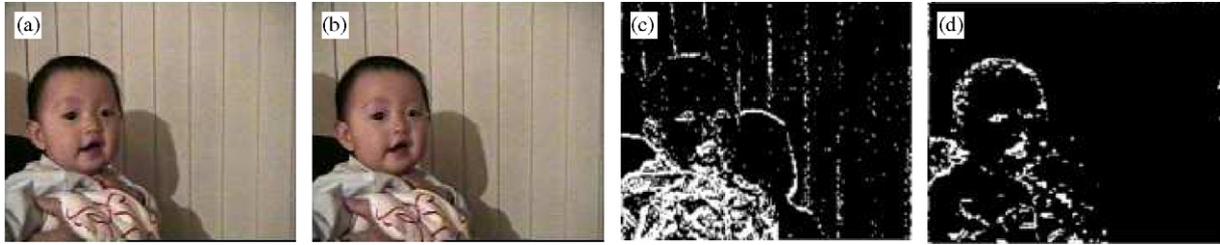


Fig. 16. Motion map: (a, b) two consecutive frames; (c) motion map by original gray-scale images; and (d) motion map by shadow-invariant images.

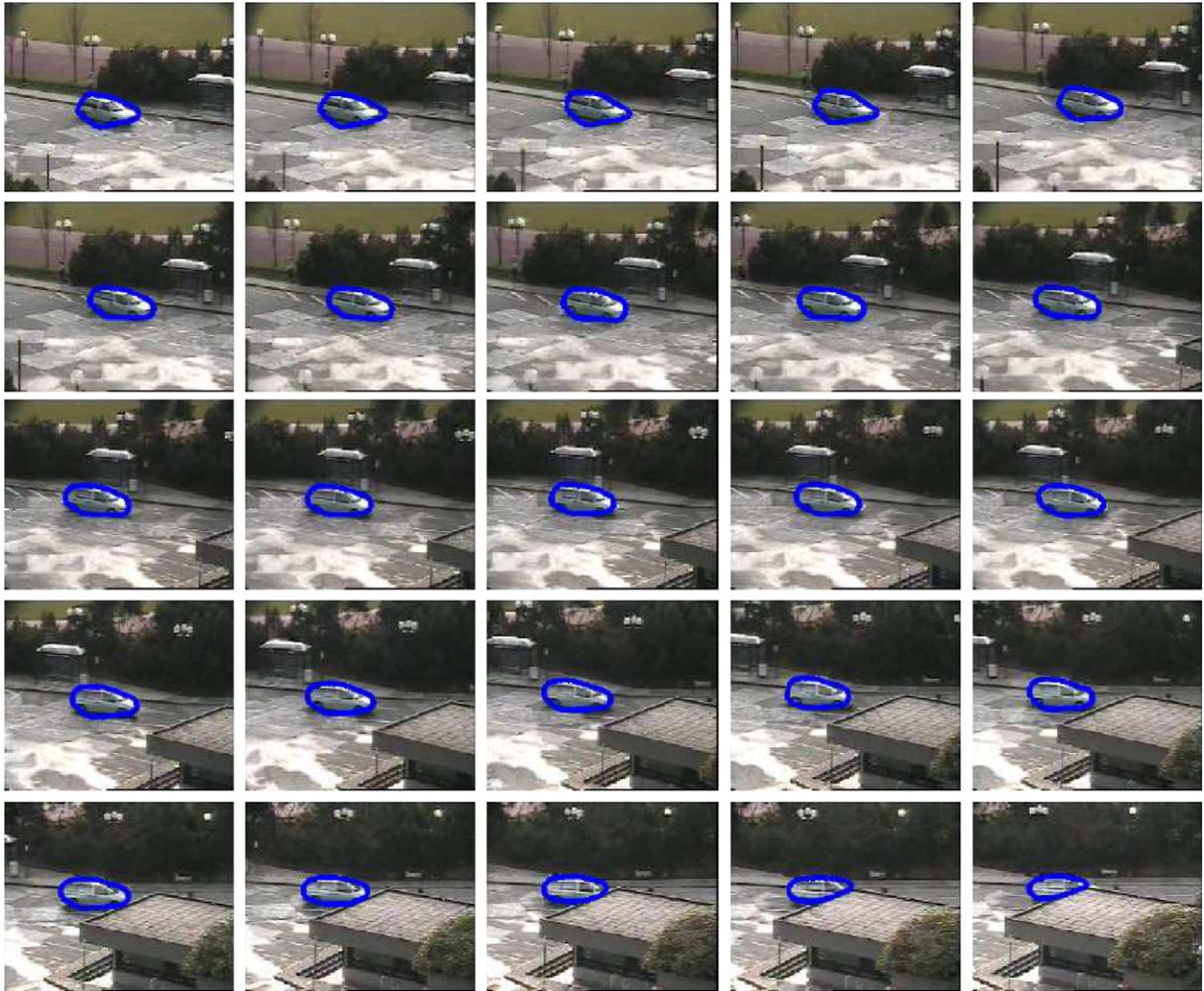


Fig. 17. Tracking result with proposed method for the car sequence, selected from a 100-frame sequence.

The inertia terms are found to be very important for the robustness of the snake. Fig. 11 shows a ten-frame test sequence. In this experiment, a slightly higher threshold is applied in the motion detection process, which results in features missing in the motion map. The motion map shows that from frame 5, motion features are missing in several successive frames. For the test sequence, we first set both inertia term I and inertia term II to zero. In this form, the method corresponds to the simple method

of initializing the snake by motion prediction. Fig. 12 shows the tracking result—the snake collapses because of the missing motion features. We then set only the shape descriptor constraint term to zero and repeat the experiment with the predictive contour constraint—the predictive contour constraint itself cannot give a robust result. The contour loses tracking at the end of the sequence, as shown in Fig. 13. One reason for the collapsing of the contour is that it is difficult to predict the contour



Fig. 18. Tracking result with proposed method for the traffic sequence, selected from a 100-frame sequence.

because of the rotation of the ball. The tracking result with both the inertia constraints for the testing video sequence is shown in Fig. 14. The inertia terms constrain the shape continuity between frames and effectively prevent the snake collapsing even in the case of missing features. Fig. 15 shows the tracking result for another longer rolling ball sequence (the video sequence has 40 frames). The result shows that the contour of the ball is well tracked without being distracted by the shadow. The contour well follows the object's contour although the scale changes.

In Fig. 16, motion is indeed present throughout, but the strong shadowing confounds a traditional tracking scheme. The shadow-removal algorithm greatly attenuates this problem, as can be seen in Fig. 16(d). In the final tracking result for this sequence, shadows do not present a problem. Figs. 17 and 18 show outdoor traffic scene tracking results. Fig. 19 shows a result for tracking a toy mouse's face in the presence of cast shadows under sunlight. Generally, while model approximation and measurement errors do affect shadow removal, shadows are usually greatly attenuated even this

happens. Furthermore, since we use a robust snake model, these factors do not cause large problems in real applications.

6. Conclusion

We present an efficient algorithm for tracking objects that is resistant to shadows. The algorithm eliminates the distracting influence from shadows and tracks the shape of the actual object. Shadow removal is based on a preliminary, simple, camera calibration. Shadow resistant tracking can be very useful for higher level vision processing such as gesture or behavior recognition. Inertia terms we introduce into the variational problem tend to preserve the object boundary shape between frames, and prevent contour collapse, with a new chordal constraint encouraging global shape preservation. The method is quite robust to moving background objects and still performs well when global motion compensation fails due to fast camera motion. This is due to the fact that the method uses color image matching in the contour prediction part, and this step will likely

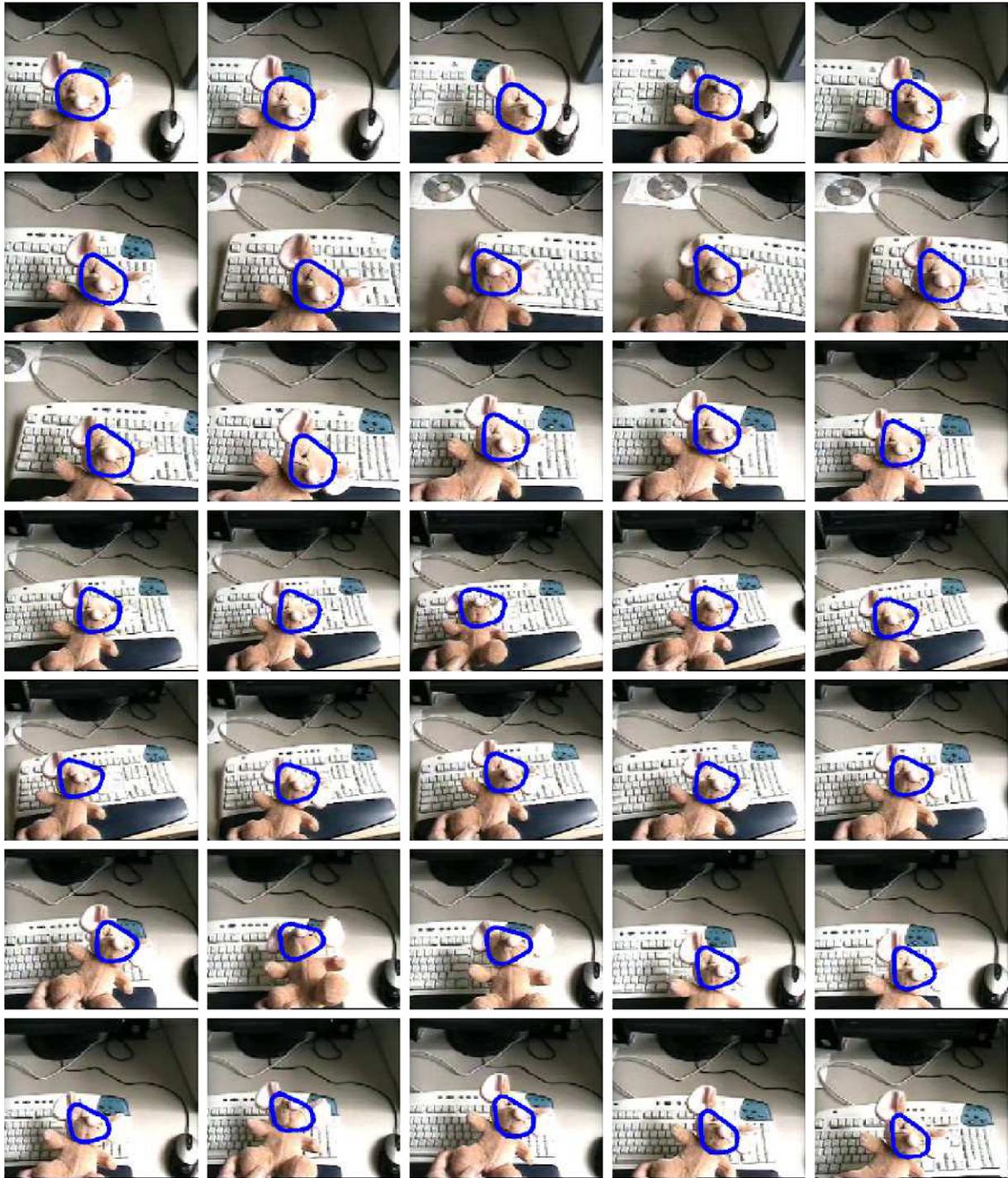


Fig. 19. Selected frames from tracking result for the 300-frame mouse sequence.

compensate for such errors. Even when global motion compensation fails, we have found that the proposed snake method still works regardless of increased background clutter in the motion map. The mouse sequence also points up a strength in the method: Since the chordal constraint is not a hard constraint, it works well even if the object deforms. In this sequence, the object's contour changes a lot in the

sequence. The proposed method still works well for these situations. Standard tracking methods such as CONDENSATION [21] needs intensive training to be successful. Another merit of the proposed method is that it does not need a training phase. For highly deformable objects, training could be useful. But there is always a trade-off between flexible constraints and robustness.

References

- [1] K. Onoguchi, Shadow elimination method for moving object detection, in: Proceedings of the 14th International Conference on Pattern Recognition, vol. 1, 1998, pp. 583–587.
- [2] Y. Sonoda, T. Ogata, Separation of moving objects and their shadows, and application to tracking of loci in the monitoring images, in: Proceedings of the 4th International Conference on Signal Processing, vol. 2, 1998, pp. 1261–1264.
- [3] E. Salvador, A. Cavallaro, T. Ebrahimi, Shadow identification and classification using invariant color models, ICASSP'01, 2001, pp. 7–11.
- [4] I. Mikic, P. Cosman, G. Kogut, M. Trivedi, Moving shadow and object detection in traffic scenes, International Conference on Pattern Recognition, vol. 1, September 2000, pp. 321–324.
- [5] J. Stauder, R. Mech, J. Ostermann, Detection of moving cast shadows for object segmentation, IEEE Trans. Multimedia 1 (1) (1999) 65–76.
- [6] S. Nadimi, B. Bhanu, Physical models for moving shadow and object detection in video, IEEE Pattern Anal. Mach. Intell. 26 (8) (2004) 1079–1087.
- [7] G.D. Finlayson, S.D. Hordley, Color constancy at a pixel, J. Opt. Soc. Amer. A 18 (2) (2001) 253–264 Also, UK Patent application no. 0000682.5. Under review, British Patent Office.
- [8] G.D. Finlayson, M.S. Drew, 4-sensor camera calibration for image representation invariant to shading, shadows, lighting, and specularities, ICCV01: International Conference on Computer Vision, IEEE, New York, 2001, pp. II: 473–480.
- [9] G.D. Finlayson, S.D. Hordley, M.S. Drew, Removing shadows from images, in: ECCV 2002: European Conference on Computer Vision, Lecture Notes in Computer Science, vol. 2353, 2002, pp. 4:823–836.
- [10] A. Blake, R. Curwen, A. Zisserman, Affine-invariant contour tracking with automatic control of spatial-temporal scale, in: Proceedings of the Fourth International Conference on Computer Vision (ICCV'93), Berlin, Germany, 1993, pp. 66–75.
- [11] D. Terzopoulos, R. Hallinan, Tracking with Kalman snakes, Active Vision, The MIT Press, Cambridge, MA, 1992, pp. 4–20.
- [12] M. Isard, A. Blake, Contour tracking by stochastic propagation of conditional density, in: Proceedings of the European Conference on Computer Vision, vol. 1, Cambridge, UK, 1996, pp. 343–356.
- [13] G. Wyszecki, W.S. Stiles, Color Science: Concepts and Methods, Quantitative Data and Formulas, second ed., Wiley, New York, 1982.
- [14] G.D. Finlayson, M.S. Drew, C. Lu, Intrinsic images by entropy minimization, in: ECCV 2004: European Conference on Computer Vision, Lecture Notes in Computer Science, vol. 3023, 2004, pp. 582–595.
- [15] M.S. Drew, C. Chen, S.D. Hordley, G.D. Finlayson, Sensor transforms for invariant image enhancement, in: Tenth Color Imaging Conference, 2002, pp. 325–329.
- [16] L.D. Cohen, I. Cohen, Finite-element methods for active contour models and balloons for 2-D and 3-D images, IEEE Trans. Pattern Anal. Mach. Intell. 15 (1993) 1131–1146.
- [17] C. Xu, J.L. Prince, Snake, shapes, and gradient vector flow, IEEE Trans. Image Process. 7 (1998) 359–369.
- [18] M. Kass, A. Wikin, D. Terzopoulos, Snakes: active contour models, Int. J. Comput. Vision 1 (1988) 321–331.
- [19] H. Jiang, M.S. Drew, Shadow-resistant tracking in video, in: ICME'03: International Conference on Multimedia and Expo, 2003, pp. III 77–80.
- [20] J. Besag, On the statistical analysis of dirty pictures, J. Roy. Statist. Soc. B 48 (3) (1986) 259–302.
- [21] M. Isard, A. Blake, CONDENSATION—conditional density propagation for visual tracking, Int. J. Comput. Vision 29 (1) (1998) 5–28.

About the Author—HAO JIANG is currently a Ph.D. candidate in the School of Computing Science at Simon Fraser University, Vancouver, Canada. His research interests are multimedia, computer vision, signal processing and communications, graphics and AI. His thesis focuses on novel approaches to energy minimization methods in computer vision and pattern recognition.

About the Author—MARK S. DREW is an Associate Professor in the School of Computing Science at Simon Fraser University, Vancouver, Canada. His background education is in Engineering Science, Mathematics, and Physics. His interests lie in the fields of multimedia, computer vision, image processing, color, photorealistic computer graphics, and visualization. He has published over 100 refereed papers in journals and conference proceedings. Dr. Drew is the holder of a US Patent in digital color processing.