

Research Article

Model-Free, Occlusion Accommodating Active Contour Tracking

Mohamed Ben Salah¹ and Amar Mitiche²

¹ *Department of Computing Science, University of Alberta, Edmonton, AB, Canada T6G 2E8*

² *INRS-EMT, National Institute of Scientific Research, Montreal, QC, Canada H5A 1K6*

Correspondence should be addressed to Amar Mitiche, mitiche@emt.inrs.ca

Received 6 September 2012; Accepted 27 September 2012

Academic Editors: C. Kotropoulos and B. Schuller

Copyright © 2012 M. Ben Salah and A. Mitiche. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

This study investigates tracking in monocular image sequences by a model-free, occlusion accommodating active contour method. The objective functional contains a model-free shape tracking term to constrain the active curve in a frame to have a shape which approximates as closely as possible the shape of the active curve in the preceding frame. It complements a kernel photometric tracking term which constrains the active curve in a frame to enclose an intensity profile that matches as closely as possible the profile within the curve in the preceding frame. This data term is in kernel form so as to forgo image modeling. The method, which is exclusively driven by the curve/level set evolution equations derived from the objective functional Euler-Lagrange conditions, can track several objects independently. Experimental validation includes examples with infrared imaging, occlusion, clutter, and articulated motion.

1. Introduction

Visual tracking consists of following moving objects through an image sequence. It is a central problem in computer vision which currently serves major applications such as scene understanding and monitoring, navigation, medical therapy, and video image analysis and synthesis.

An image object can be tracked according to its photometric appearance or its shape. Appearance-based tracking uses the object intensity profile which it assumes is distinctive of the object against its background. Shape tracking uses a geometric description of the object modulo, a class of relevant deformations. When an object is imaged from several distinct viewpoints, three-dimensional pose information can be recovered to assist tracking [1]. However, photometric appearance and shape remain the basic visual support to locate the same object from one image to the next in a sequence.

To track an image object requires detecting it initially. Detection is traditionally done by successive frame pointwise differencing [2] followed by filtering of the difference image to extract blob parts of the moving object. In general, processing the blobs is tedious and unyielding beyond

circumscribing the moving object within figures such as rectangles or ellipses. Such bounding figures meet the need of applications where tracking is to follow the object grossly placed about a mean position [3, 4]. However, they are largely insufficient when it is necessary to identify the object or interpret its behavior. An object contour can be formally described by a closed plane curve which bounds a region of a given appearance and satisfies some shape constraints. These constraints can be embedded in a functional which is minimized by the curve which coincides with the desired object boundary. The most common shape constraint is regularity, or smoothness, conveyed by a curve length term in the objective functional. When available, the boundary geometry can also be accounted for by a shape prior in the functional [5–7]. The image appearance of the region bounded by an object boundary is conveyed by a data term in the functional. The most common such term measures the conformity of the observed intensity to a model distribution [8]. The minimization of the functional results in evolution equations which drive a variable curve, called an active contour, to coincide with the boundary of the desired object. This variational active contour formalism has been extensively used in image segmentation [8]. It has resulted

in efficient and accurate algorithms whose behaviour can be clearly explained by the curve evolution equations. A few studies have also used active contours which also been used in tracking [9–13].

Tracking has been first addressed using active curves/level sets using geodesic functionals where some tracking terms are generally driven by the image gradient while detection terms are driven by the distribution of the interframe image difference at motion boundaries. Neither of such terms has a provision for occlusion, so that only objects which move in full view can be processed. In this setting, detection is commonly performed to completion prior to tracking by positioning an initial curve close to the image domain boundary, so that it contains the moving object, and making it move inward. The initial curve is so placed because the direction of motion of a geodesic curve is constant, inward or outward. Detection assumes a relatively uniform background and a strong motion boundary contrast to prevent the geodesic curve to move past the moving object boundary. A leak would otherwise cause the curve to ultimately vanish. Because tracking follows detection, the detected curve must contain the object of interest since tracking is mediated by a (inward moving) geodesic. When the tracking term has closed up on the object, the detection process is restarted with a curve open wide enough to circumscribe the object.

The region-based active curve/level set method of [9] alleviates the geodesic major problems of curve leaking through weak contrast motion boundaries and unidirectional curve evolution. However, the method uses a background model learned a priori and, therefore, is applicable only where such knowledge can be acquired.

A photometric appearance model can be used to selectively detect a moving object and then track it. This is how [10] stated the problem, using the image empirical distribution (histogram) to describe the photometric appearance of the moving object. The formalism has been adapted to medical image sequences in [14]. The method is applicable to a single object and the motion of the object must be slow enough so that the detected contour at a given instant of time is close to the object contour at the next instant. Histograms may code the object boundary inaccurately because they are obtained by summing up the image over the object region. This would be of no consequence where the object need not be identified nor its behaviour interpreted.

When a model of the moving object boundary shape is available, it can assist tracking. This has been demonstrated in [11]. First, a training set of distinct outlines of the object, resulting from various viewing perspectives or distances, is used to learn a geometric shape description of the moving object by principal component analysis (PCA). PCA retains of the boundary full string of descriptive points only a few which are the most significant to characterize the boundary shape. These few principal components, which economically define a class of shapes, are embedded in a shape prior of an active curve/level set formulation. Shape priors have been investigated in several studies in image segmentation [5–7, 15] but scarcely in tracking. To further accommodate distortions caused by noise, clutter, and occlusions, tracking

continually updates the shape prior (termed a dynamic prior) using a number of the previously observed contours in the tracking trail. The data term plays a lesser role and it simply assumes the intensity of both the object and the background to be Gaussian, an assumption that is not generally applicable.

Shape priors have also been used in [12] for multiobject contour tracking with occlusion handling. The photometric appearance, which fuses color and texture, is modeled by a mixture of Gaussians. The formulation does not activate the appearance and shape functionals concurrently. The former is used for detecting the moving object when there is no occlusion. The latter is considered only when an occlusion starts. An independent external process detects occlusion when it occurs between any two tracked objects, but occlusions involving a tracked object and a static or a nontracked moving object are not processed.

The study in [13, 16] has investigated tracking along a quite different vein. For a moving object region at some instant, the corresponding region at the next instant is the one that best matches its intensity profile according to an integral functional of best pointwise matches. Although the pointwise match function uses an image neighborhood bounded by the maximum expected pointwise displacement of the object, there is no actual computation of motion. The formulation can embed efficient object image models in a simple manner. The method processes single objects. It has no mechanism to address occlusion, via a shape model, for instance, and the neighborhood operations can lose some identifying detail along the object contour. Under the method assumptions, an implementation has shown a sturdy behaviour over long sequences.

The subject of our study is model-free, occlusion accommodating contour tracking in an image sequence. Although contour tracking from one image to the next in a sequence can be viewed as a particular problem in image segmentation, namely, determining in the next image a single region described by an intensity profile or by the shape of its boundary, it has three fundamental idiosyncrasies in general application: (1) successive contour deformations, even when no single one is substantial, can cumulate to alter the contour shape out of any prior shape model, (2) sections of the contour can be occluded unpredictably, and (3) several objects can be targeted for tracking simultaneously. These idiosyncrasies of general tracking are the main concern of our study.

In controllable settings, models of contour shape or of allowable contour deformations can be learned and contribute subsequently to locate a particular object boundary. However, this is not always possible or practicable. For instance, training data may be unavailable or inaccessible, or tracking may target objects which can be occluded, or undergo profile variations during motion too complex to circumscribe in a single model. The examples we used show some of these situations: Figure 1 illustrates the case of tracking a vehicle backing up toward the camera and (Figure 2) a differently shaped vehicle receding, and the same tracking formulation is to adapt to other vehicle shapes as well. Figures 7 and 8 show movements which cause

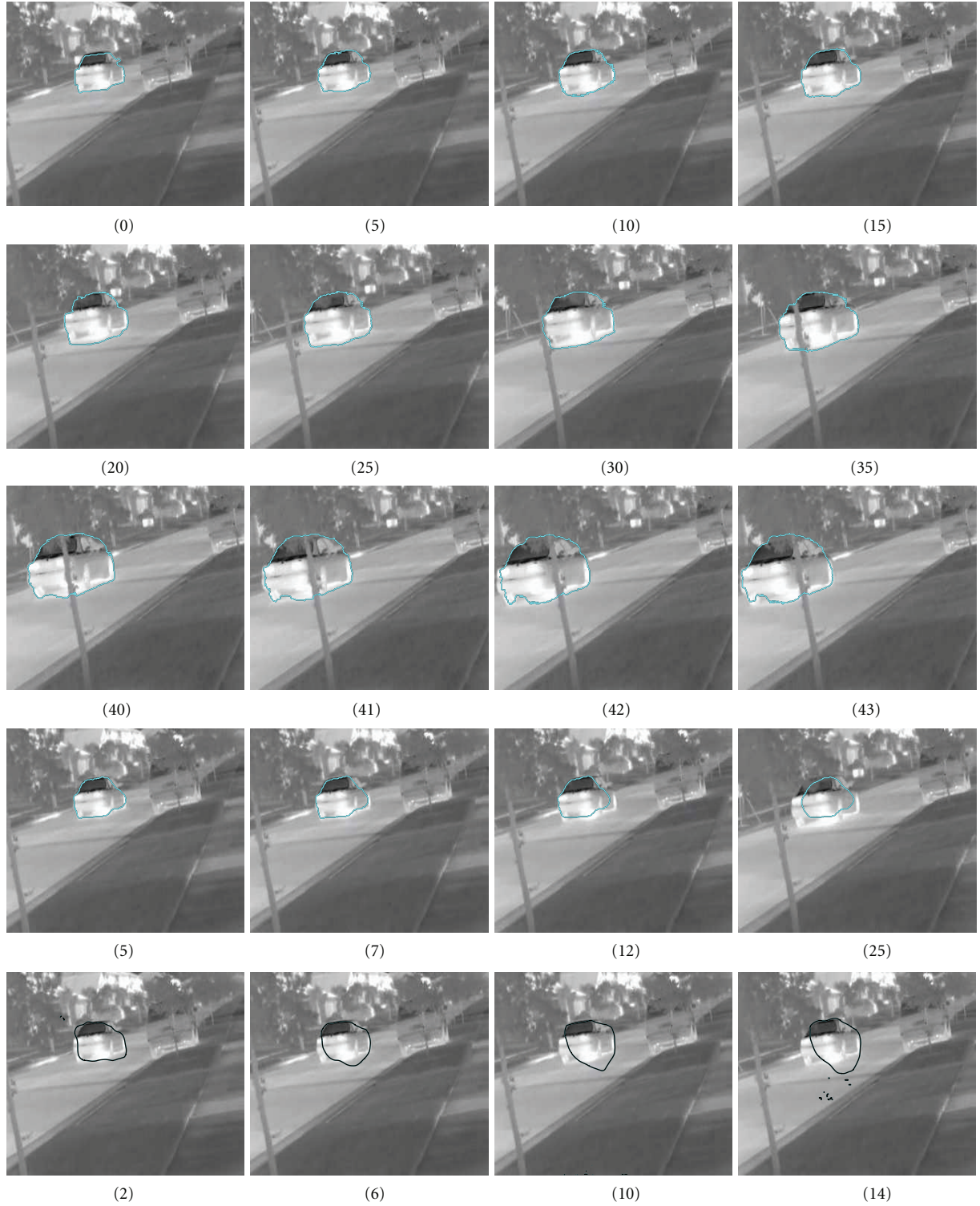


FIGURE 1: Tracking in an infrared (IR) image sequence. The contrast along the moving boundary is weak, particularly in the portion between the car and the street. There is also partial occlusion behind the tree toward the end of the sequence. *Top three rows*: tracking with the proposed kernel-based intensity tracking term. *Fourth row*: tracking in [13] with the Gaussian assumption. *Bottom row*: tracking by distributions with the method in [10]. Under each image is the frame number.

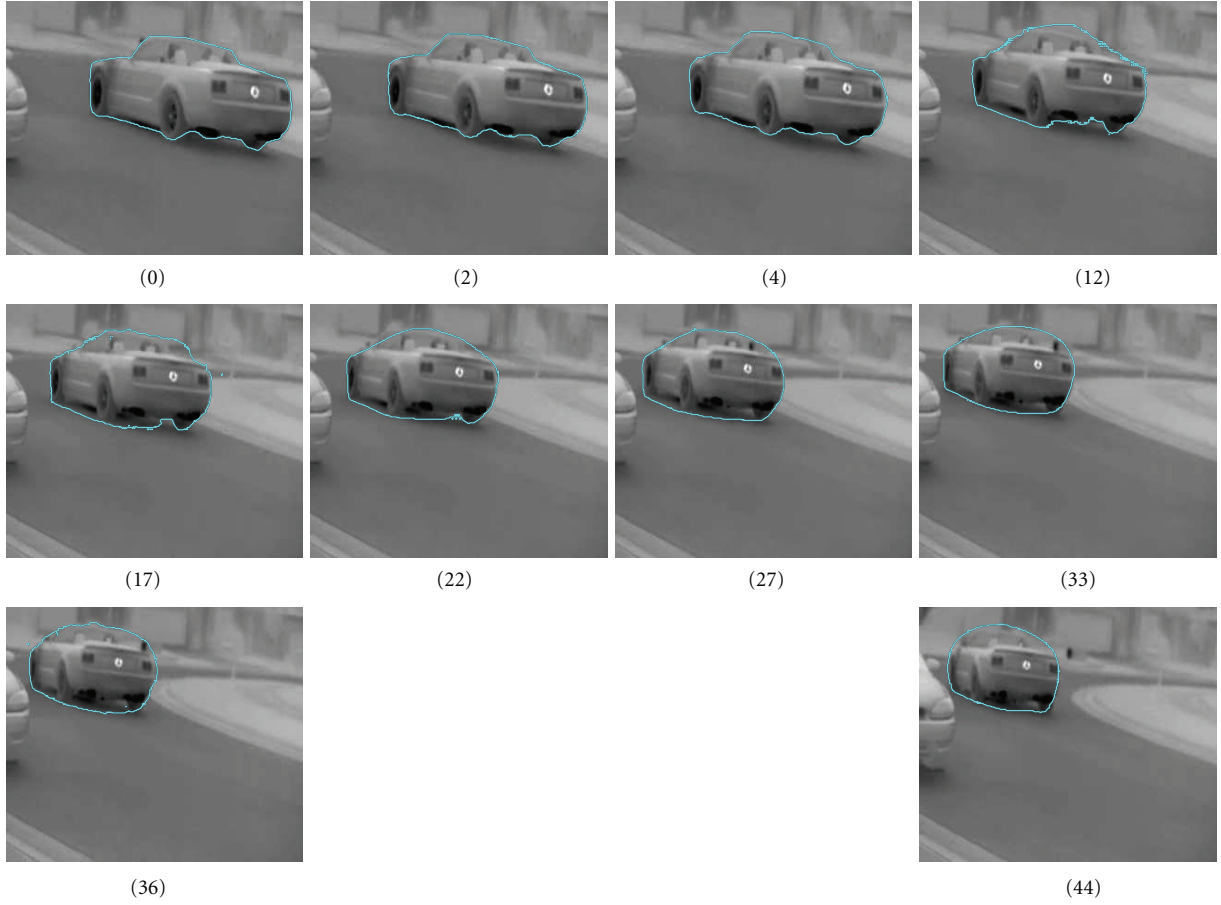


FIGURE 2: Tracking of an IR sequence. The active curve has been able to adapt to the changing size and outline of the object.

the objects profile to vary considerably so that the ensemble of the sequence profiles would poorly fit a single model description. Figure 3 shows an example of occlusion as can occur in reality and Figure 6 depicts a case of multiple object tracking. Designed to generalize ordinary geometric models to effective general representations in settings where learning object shape appearance is possible and practicable, the method in [11, 17] is not applicable in such situations as depicted by these examples. A more accessible model would be the contour template drawn from an exemplar object and subsequently used to track from one image to the next via a general affine transformation. However, such model and strategy are notoriously inapt at tracking in most situations.

Our purpose is model-free contour tracking rather than contour tracking via modeling, applicable in such situations as depicted in the experimentation examples, occlusion and multiobject tracking for instance. The method we investigate does this by taking the object photometric and shape appearance in one frame as the guide to locate the object in the next. This natural strategy is simple and effective. The use of the contour shape in the previous frame as a geometric model for the contour in the current frame decomposes a possibly significant deformation occurring over an extended period

of time into a sequence of generally simpler in-between frame deformations, affording the possibility to track in the presence of large cumulative shape alteration. This is of considerable benefit when modeling object shape variations is not possible or practicable. The object photometric and shape tracking constraints are each embedded in an objective functional and resolved by Euler-Lagrange equations via level sets. The photometric tracking term is of the form in [13, 16] but expressed in a kernel induced space [18–20]. This kernel expression forgoes the need for image data models, which can be a significant advantage when the photometric appearance of the objects to track is not known beforehand or is too disparate to model reliably. The purpose of the shape tracking term is to assist tracking in the presence of partial occlusion, whether the occluding object is static or mobile.

The method is exclusively driven by the curve/level set evolution equations; no other processes intervene. Against the background of contour tracking methods, the scheme accumulates the following advantages: (1) it is model-free, (2) it accommodates occlusion, (3) it is applicable via a kernel mapping to image sequences of different modalities, and (4) it is applicable to multiple-object tracking.

The remainder of this paper is organized as follows. Section 2 describes the proposed tracking energy. Section 2.1

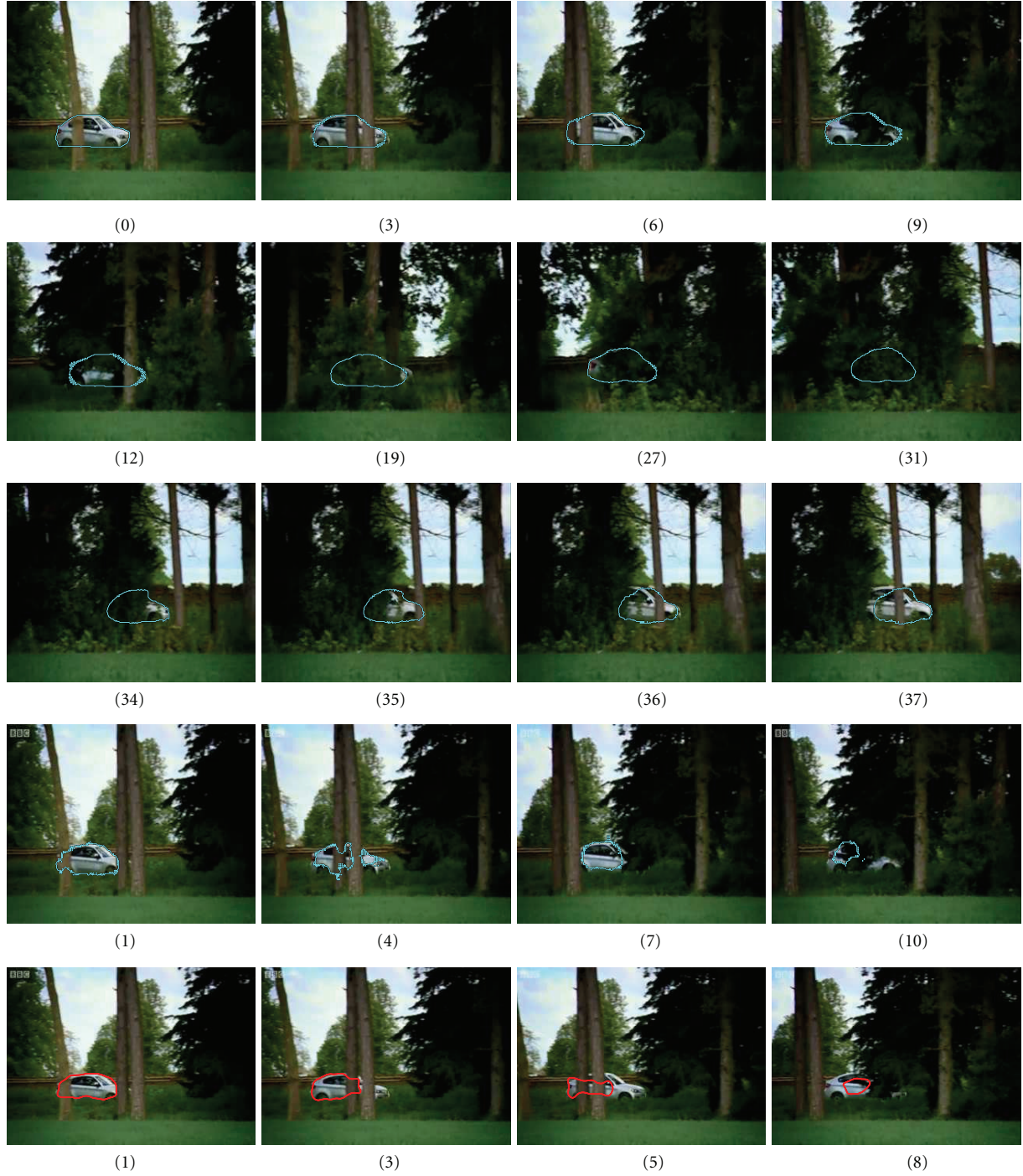


FIGURE 3: Tracking on the *Car-behind-trees* sequence with total occlusion. The car velocity is constant during the time of observation. Significant partial occlusion and total occlusion occur in the sequence. *Top three rows*: tracking with the proposed method. During total occlusion, the shape tracking term predominates to guide contour evolution. After total occlusion (third row) the car is detected although the detected silhouette is inaccurate. *Fourth row*: tracking with the method in [13]. *Last row*: tracking with the method in [10].

develops the kernel-based intensity tracking term and Section 2.2 the shape tracking term. Section 3 derives the level set representation of the objective functional and corresponding Euler Lagrange descent. Section 4 provides experiments for proof of concept and comparisons. Finally, Section 5 contains a conclusion.

2. Formulation

Let $(I_n)_n$, where n is a nonnegative integer representing time instants, be a sequence of images defined on a common open domain Ω . Assume that we are given a moving region $\mathbf{R}_0 \subset \Omega$ in the image, I_k , at instant k . We estimate its corresponding

region, \mathbf{R}_1 , in the next image, I_{k+1} , by minimizing an objective functional, \mathcal{E} , which is the sum of a *photometric tracking* term, $\mathcal{E}_{\mathcal{P}}$, and a *shape tracking* term, $\mathcal{E}_{\mathcal{S}}$:

$$\begin{aligned} \mathcal{E}(\mathbf{R} \mid I_k, I_{k+1}, \mathbf{R}_0) &= \mathcal{E}_{\mathcal{P}}(I_{k+1} \mid I_k, \mathbf{R}_0, \mathbf{R}) + \mathcal{E}_{\mathcal{S}}(\mathbf{R} \mid \mathbf{R}_0), \\ \mathbf{R}_1 &= \arg \min_{\mathbf{R}} \mathcal{E}(\mathbf{R} \mid I_k, I_{k+1}, \mathbf{R}_0). \end{aligned} \quad (1)$$

Such a problem has been stated in [13] using a MAP estimation argument as in [21]. The photometric tracking term will bias the solution toward a region in I_{k+1} whose intensity profile closely matches the one of \mathbf{R}_0 and the shape tracking term will look for a region whose boundary closely resembles the boundary of \mathbf{R}_0 . The photometric term uses the image transformed by a kernel function so as to accommodate various image models without recourse to model learning. The shape tracking term, also model-free, assists tracking and accommodates occlusion.

2.1. Kernel-Based Photometric Tracking Term. Following [13], we will use a photometric tracking term which embeds a pointwise correspondence between I_{k+1} and I_k . The basic premise is that there exists a one-to-one mapping between \mathbf{R}_0 and \mathbf{R}_1 , hence between \mathbf{R}_0^c and \mathbf{R}_1^c , and that the image at corresponding points is the same up to additive noise. Gaussian noise leads to a data term which is the squared image difference at corresponding points integrated over a region and its complement [13, 16]. However, the Gaussian model is not generally applicable [8]. To allow more general models, we use the following photometric tracking term:

$$\begin{aligned} \mathcal{E}_{\mathcal{P}}(I_{k+1} \mid I_k, \mathbf{R}_0, \mathbf{R}) &= \int_{\mathbf{R}} \inf_{\{l: \|l\| \leq \delta, \mathbf{x}+l \in \mathbf{R}_0\}} (\phi(I_{k+1}(\mathbf{x})) - \phi(I_k(\mathbf{x}+l)))^2 d\mathbf{x} \\ &+ \int_{\mathbf{R}^c} \inf_{\{l: \|l\| \leq \delta, \mathbf{x}+l \in \mathbf{R}_0^c\}} (\phi(I_{k+1}(\mathbf{x})) - \phi(I_k(\mathbf{x}+l)))^2 d\mathbf{x}, \end{aligned} \quad (2)$$

where $\phi(\cdot)$ is a nonlinear mapping of the original image data into a space of higher dimension (called the kernel Hilbert space) where the Gaussian model becomes applicable. This mapping is done implicitly via a kernel function [22, 23] as explained subsequently. Kernel methods have been very effective in image segmentation [19, 20] and image denoising [18] as well as data clustering [24]. According to the Mercer theorem [22], there is no need to know function ϕ explicitly because the dot product in the space of the transformed data can be expressed using a continuous, symmetric, positive semidefinite kernel function. The kernel function, $K(\mathbf{y}, \mathbf{z})$, verifies

$$K(\mathbf{y}, \mathbf{z}) = \phi(\mathbf{y})^T \cdot \phi(\mathbf{z}), \quad \forall (\mathbf{y}, \mathbf{z}) \in \mathbb{R}^2, \quad (3)$$

where “ \cdot ” is the dot product in the feature space. Application of this *kernel trick* [22, 23] to $\|\phi(I_{k+1}) - \phi(I_k)\|^2$ gives

$$\begin{aligned} D_K(I_{k+1}, I_k) &\equiv \|\phi(I_{k+1}) - \phi(I_k)\|^2 \\ &= (\phi(I_{k+1}) - \phi(I_k))^T \cdot (\phi(I_{k+1}) - \phi(I_k)) \\ &= \phi(I_{k+1})^T \cdot \phi(I_{k+1}) - \phi(I_k)^T \cdot \phi(I_{k+1}) \\ &\quad - \phi(I_{k+1})^T \cdot \phi(I_k) + \phi(I_k)^T \cdot \phi(I_k) \\ &= K(I_{k+1}, I_{k+1}) + K(I_k, I_k) - 2K(I_{k+1}, I_k). \end{aligned} \quad (4)$$

In terms of D_K , which is a non-Euclidian distance in the image data space, (2) is written as

$$\begin{aligned} \mathcal{E}_{\mathcal{P}}(I_{k+1} \mid I_k, \mathbf{R}_0, \mathbf{R}_1) &= \int_{\mathbf{R}} \inf_{\{l: \|l\| \leq \delta, \mathbf{x}+l \in \mathbf{R}_0\}} D_K(I_{k+1}(\mathbf{x}), I_k(\mathbf{x}+l)) d\mathbf{x} \\ &+ \int_{\mathbf{R}^c} \inf_{\{l: \|l\| \leq \delta, \mathbf{x}+l \in \mathbf{R}_0^c\}} D_K(I_{k+1}(\mathbf{x}), I_k(\mathbf{x}+l)) d\mathbf{x}. \end{aligned} \quad (5)$$

Several kernels are available in practice [23]. We will use the radial basis function (RBF) [24, 25], a kernel which has been very effective in image segmentation [19, 20]:

$$K(\mathbf{y}, \mathbf{z}) = \exp\left(\frac{-\|\mathbf{y} - \mathbf{z}\|^2}{\sigma^2}\right), \quad \sigma > 0. \quad (6)$$

2.2. Shape Tracking Term. We represent the object boundary by a closed regular plane curve \mathcal{C} , defined implicitly as the zero level set of an embedding function $\Phi : \mathbb{R}^2 \rightarrow \mathbb{R}$ [26], with the convention that the interior of \mathcal{C} corresponds to $\Phi > 0$. The level set representation has been used almost systematically for active contours in computer vision problems [27–29] such as image segmentation [8] and tracking [9, 10, 16]. Among other advantages, the level set representation allows the topology of an evolving curve to change and can be implemented by numerically stable schemes. We will use the evolving contour-signed distance function (SDF) as its level set function. In this case, $\Phi(\mathbf{x})$ is the smallest distance from \mathbf{x} to the contour, positive inside and negative outside the region bounded by the contour.

Let Φ_0 be the signed distance level set function of the object of interest at time k and $\mathbf{R}_0 = \{\mathbf{x} \in \Omega \mid \Phi_0(\mathbf{x}) > 0\}$. We will use Φ_0 as a shape prior at time k to estimate the region \mathbf{R}_1 corresponding to \mathbf{R}_0 at the next instant $k+1$. Therefore, the shape tracking term, $\mathcal{E}_{\mathcal{S}}$, will measure the similarity between the evolving minimization SDF Φ and the SDF Φ_0 . Several such similarities are available [6, 15, 30]. We will use the *displaced SDF difference* distance:

$$d^2(\Phi, \Phi_0) = \int_{\Omega} (H(\Phi(\mathbf{x})) - H(\Phi_0(\mathbf{x} + \mathbf{h})))^2 d\mathbf{x}, \quad (7)$$

where $H(\cdot)$ denotes the Heaviside step function and \mathbf{h} is the displacement vector field between the frames at times k and $k+1$. This field can be included as a variable in the objective functional. However, this is not necessary in tracking.

Instead, it is sufficient to approximate \mathbf{h} by the average SDF difference between the previous two frames because the functional minimization with both the photometric and the shape terms will further seek to align the evolving and the previous SDFs. If we also impose regularity to the evolving contour, using curve length, the shape tracking term at time k is given by

$$\mathcal{E}_s(\mathbf{R} \mid \mathbf{R}_0) = \frac{\lambda_{sp}}{2} d^2(\Phi, \Phi_0) + \lambda_L \oint_{\partial \mathbf{R}} ds, \quad (8)$$

where λ_{sp} and λ_L are positive reals, and $\mathbf{R} = \{\mathbf{x} \in \Omega \mid \Phi(\mathbf{x}) > 0\}$.

Using the photometric and the shape tracking terms (5) and (8), the tracking functional is

$$\begin{aligned} \mathcal{E}(\mathbf{R} \mid I_k, I_{k+1}, \mathbf{R}_0) &= \int_{\mathbf{R}} \inf_{\{l: \|l\| \leq \delta, \mathbf{x}+l \in \mathbf{R}_0\}} D_K(I_{k+1}(\mathbf{x}), I_k(\mathbf{x}+l)) d\mathbf{x} \\ &+ \int_{\mathbf{R}^c} \inf_{\{l: \|l\| \leq \delta, \mathbf{x}+l \in \mathbf{R}_0^c\}} D_K(I_{k+1}(\mathbf{x}), I_k(\mathbf{x}+l)) d\mathbf{x} \quad (9) \\ &+ \lambda_L \oint_{\partial \mathbf{R}} ds + \frac{\lambda_{sp}}{2} d^2(\Phi, \Phi_0). \end{aligned}$$

3. Functional Minimization

The objective functional (9) can be minimized by rewriting it terms of the level set function Φ and deriving the corresponding Euler-Lagrange equations which will give the evolution equations of Φ .

3.1. Level Set Evolution Equations. The tracking functional (9) is rewritten in terms of the level set function Φ as

$$\begin{aligned} \mathcal{E}(\Phi) &= \int_{\Omega} H(\Phi) \inf_{\{l: \|l\| \leq \delta, \mathbf{x}+l \in \mathbf{R}_0\}} D_K(I_{k+1}(\mathbf{x}), I_k(\mathbf{x}+l)) d\mathbf{x} \\ &+ \int_{\Omega} (1 - H(\Phi)) \inf_{\{l: \|l\| \leq \delta, \mathbf{x}+l \in \mathbf{R}_0^c\}} D_K(I_{k+1}(\mathbf{x}), I_k(\mathbf{x}+l)) d\mathbf{x} \\ &+ \lambda_L \int_{\Omega} |\nabla H(\Phi)| d\mathbf{x} \\ &+ \frac{\lambda_{sp}}{2} \int_{\Omega} (H(\Phi(\mathbf{x})) - H(\Phi_0(\mathbf{x} + \mathbf{h})))^2 d\mathbf{x}, \quad (10) \end{aligned}$$

which gives the following level set evolution equations for its minimization [8]:

$$\begin{aligned} \frac{\partial \Phi}{\partial \tau} &= - \frac{\partial \mathcal{E}(\Phi)}{\partial \Phi} = -\delta(\Phi) \\ &\times \left[\xi_1(\mathbf{x}) - \xi_2(\mathbf{x}) + \lambda_L \operatorname{div} \left(\frac{\nabla \Phi}{|\nabla \Phi|} \right) \right. \\ &\quad \left. + \lambda_{sp} (H(\Phi(\mathbf{x})) - H(\Phi_0(\mathbf{x} + \mathbf{h}))) \right], \quad (11) \end{aligned}$$

where $H\Phi \equiv H(\Phi)$ (for notational convenience), τ is the algorithmic time, and δ is the Dirac distribution, and

$$\begin{aligned} \xi_1(\mathbf{x}) &= \inf_{\{l: \|l\| \leq \delta, \mathbf{x}+l \in \mathbf{R}_0\}} D_K(I_{k+1}(\mathbf{x}), I_k(\mathbf{x}+l)) d\mathbf{x}, \\ \xi_2(\mathbf{x}) &= \inf_{\{l: \|l\| \leq \delta, \mathbf{x}+l \in \mathbf{R}_0^c\}} D_K(I_{k+1}(\mathbf{x}), I_k(\mathbf{x}+l)) d\mathbf{x}. \end{aligned} \quad (12)$$

The behaviour of the algorithm when an occlusion sets in is as follows. Consider the object is visible at the current instant and a moderate partial occlusion occurs at the next. In the occluded segment of the object, there will be a competition for points between the object and the background as follows: for a point in this segment, the term $\xi_1 - \xi_2$ will likely be negative because the point intensity will likely match better that of a point in the background than in the object. In such a case, and given the meaning of distance functions Φ , Φ_0 , and displacement \mathbf{h} , $H\Phi - H\Phi_0$ will be 1 and the shape term will be equal to λ_{sp} . Omitting the smoothness term, the point will be claimed by the background if λ_{sp} is smaller than $\xi_1 - \xi_2$. Otherwise it will be claimed by the object. Therefore, for a value of λ_{sp} sufficiently large, the point will be claimed by the object as desired. A good value for the algorithm to behave in this desired way is determined empirically. However, a value determined for one or a few sequences in an application generally remains fixed for the application. Adjusting the value for applications where occlusion is moderate between frames, that is, where it occurs progressively, is not an arduous task. What is perhaps more important to bear in mind is that a properly resolved occluded segment of the object at some instant will become “part” of the object at the next and will likely continue to be resolved properly. The occluding object region will move out of the object when the occluded part of the object reappears progressively. The algorithm behaviour is, of course, similar when an occluded part of the object is progressively uncovered. Notwithstanding its ability to keep track of objects in their movement during progressive occlusion, the scheme has no explicit provision to recover portions lost to tracking, that is, which are visible but unrecorded by the tracking process. Also, as a general tracking process, it can be overwhelmed by sudden significant occlusion activity.

3.2. Generalizations. The method can be used for intensity as well as vector-valued images such as color image sequences. The generalization to vector-valued images is straightforward and, of course, affects only the photometric term of the tracking functional [19, 20]. The generalization to multiple object tracking is also straightforward. Consider the problem of tracking N objects in a sequence and let Φ^i , $i = 1, \dots, N$, be N different level set functions, each affected to a different object. Then each level set is simply evolved independently using evolution equations as in (11). The experimental section shows examples for both vector-valued image sequences and multiple-object tracking.

4. Experimentation

The purpose of the following experimentation is to demonstrate the validity of the formulation and its implementation. The experiments have been carefully chosen so as to serve as proof of concept and be relevant for an informative comparison to state-of-the-art contour tracking methods. Each example stresses a particular aspect of the algorithm: there are examples with infrared images to show the relevance of using a kernel function in the photometric tracking term; examples with occlusion and clutter to show the relevance of the shape tracking term; examples with simultaneous multiobject tracking and shape altering articulated motion. The comparisons, on pertinent sequences, are made with the tracking algorithm without motion computation of [13] and the nonparametric distribution tracking algorithm of [10], both discussed in the literature review of Section 1.

All of the experiments focus on tracking proper. However, by definition, tracking must be initially provided with an object to track. Object detection has been addressed using a variety of means such as geodesics evolved by motion sensitive image statistics and image segmentation by region statistics [31]. Because the subject of this study is tracking proper, initialization was expeditiously treated semimanually as follows: a contour is placed approximately but near the outline of the object to track and made to move for a better initialization by a few iterations of the algorithm using the photometric tracking term alone.

4.1. Infrared (IR) Sequences. A recorded infrared (IR) signal is typically of low spatial resolution, low contrast, severely noisy, and variant with atmospheric conditions [32]. The sensor has built-in postprocessing algorithms and architectures to produce a visually meaningful image sequence. However, the processed images remain of low resolution and low contrast and carry distortions which are difficult to characterize or model. The purpose of the following IR examples is to show the pertinence of the kernel transform in the photometric term when tracking in the presence of such complex non-Gaussian noise. Accordingly, the algorithm was run without the shape tracking term ($\lambda_{sp} = 0$) so as to show exclusively the role of the kernel photometric tracking term. δ has been set to 10 pixels.

In this first example (Figure 1), a car is filmed backing up in a neighborhood street. This sequence presents two main difficulties. First, the contrast along the moving boundary is weak, particularly in the portion between the car and the street. The larger filming distance at the beginning of the sequence causes elevated noise which weakens the moving boundary definition. The second difficulty is caused by the partial occlusion behind the tree toward the end of the sequence. The first three rows of Figure 1 show the behaviour of the algorithm with the kernel photometric tracking term. Tracking has followed closely the moving boundary throughout the sequence, even though the upper portion is approximate toward the end. The fourth row of Figure 1 depicts the application of the photometric term without a kernel transformation of the image as in [13]; in this case, tracking has lost the moving boundary by the 12th frame.

The last row of Figure 1 shows the results corresponding to the histogram matching method in [10]. In general, histograms would not code object boundaries accurately when the intensity distributions inside and outside the object region overlap substantially, which is the case with this sequence. As a result, the active contour tracks only grossly the object, missing some of its parts and including portions from the background.

The second example, depicted in Figure 2, further documents the behaviour of the kernel photometric tracking term on an IR sequence. One obvious difficulty with the sequence is the faint contrast at the boundary between the image of the moving car and the upper background, particularly toward the end of the sequence when the image of the car is so small as to expose a major portion of its boundary to the similarly looking upper background. Another obvious difficulty is the variation in the tracked shape (its size and outline) during its run through the sequence. The active contour has kept close to the moving car boundary; it adhered most of the time although its position becomes approximate toward the end of the sequence.

The next examples will take up the role of the shape tracking term. This term is notably useful to assist tracking in the presence of occlusion.

4.2. Occlusion. The shape tracking term assists tracking because it constrains the active curve in a frame to have a shape which approximates as closely as possible the shape of the active curve in the preceding frame. This term complements the photometric tracking term which constrains the active curve in a frame to enclose an intensity profile that matches as closely as possible the profile within the curve in the preceding frame. The role of the shape tracking term becomes essential with occlusion, which often occurs when viewing moving objects. A moving object can be occluded partially or totally by another object and self-occlusion can also occur, particularly with articulated objects.

The *Car-behind-trees* sequence, depicted in Figure 3, shows a car driven behind a row of trees. Partial occlusions occur repeatedly and there is total occlusion towards the end of the sequence before the car reappears. When occlusion occurs there is less relevant information available to the photometric correspondence process to assist tracking. The contribution of the photometric tracking term to the minimization of the objective functional decreases and that of the shape tracking term increases, the more so with more extensive occlusions. Therefore, the role of the shape tracking term which, we recall, acts via the displaced SDF difference, becomes essential. This is illustrated in Figure 3 which shows the results of tracking with the shape tracking term and without. With the shape tracking term (the top three rows of Figure 3), the active contour delineates correctly the moving car during partial occlusion by recovering its hidden parts. During total occlusion, the curve evolves essentially to keep as closely as possible its shape in the preceding frame, although the exact shape is elusive in such cases. The fourth row of Figure 3 shows the results obtained using

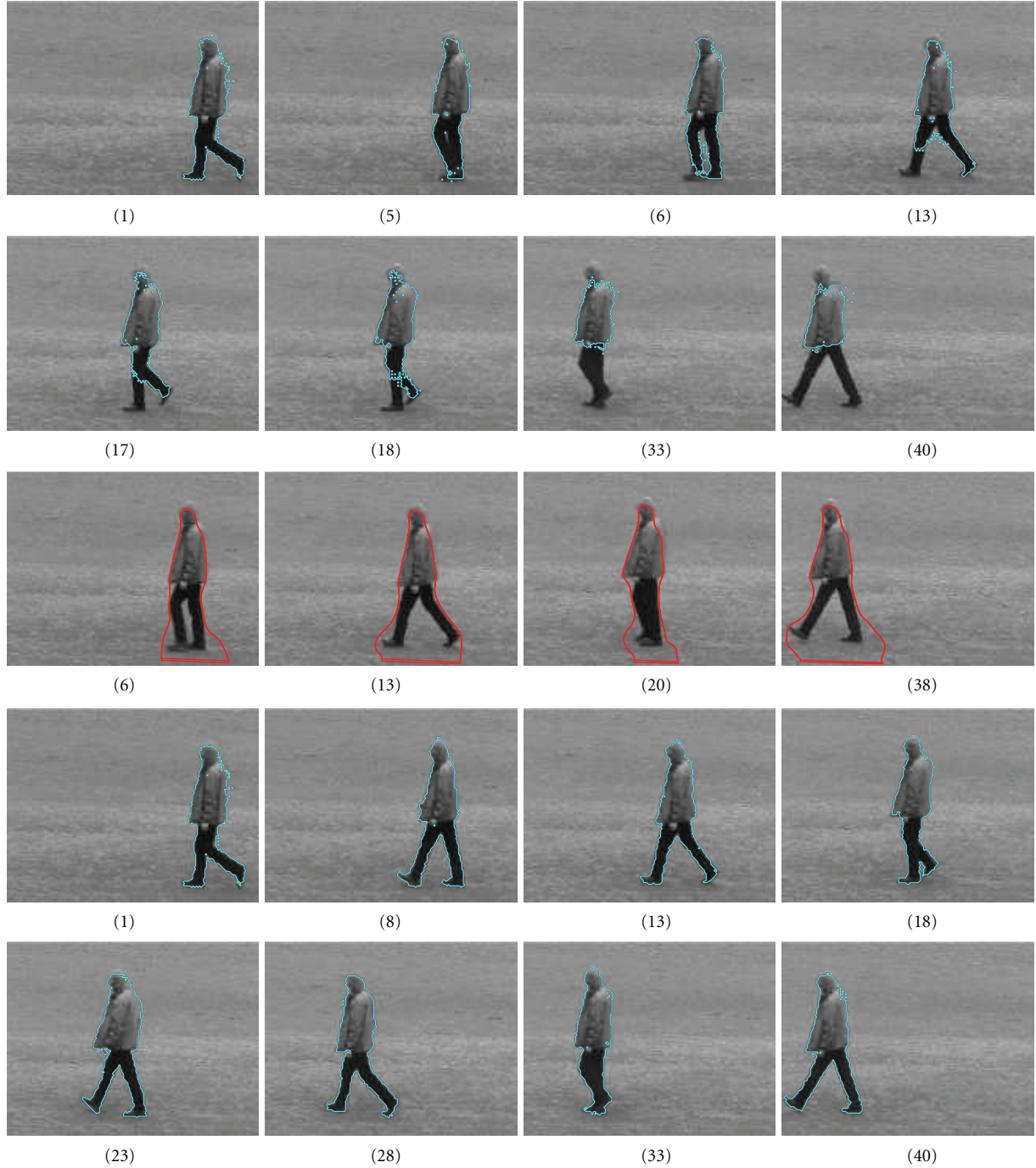


FIGURE 4: Tracking on the grey scale sequence *Walk*. *Top two rows*: method in [13]: the active curve encroaches on the object region though the blur between the walker feet and the grassy ground and continues drifting in due to the smoothness term and the uncertain outcome of intensity matching within the object uniform area. *Third row*: tracking with the method in [10]: the region intensity histogram representation allowed only a gross outline of the object. *Bottom two rows*: proposed method: by constraining the curve to retain a similar shape between consecutive frames, encroachment on the object and subsequent movement inward are averted.

the algorithm without the shape tracking term. Tracking was confused early in the sequence and lost the object by the 10th frame, without recovery in subsequent frames. The bottom row shows the results obtained by applying the method in [10]. With partial occlusion, the contour has kept the track of the appearing part of the car. After that, the contour has lost

completely the target and disappeared from the scene right after frame number 9.

This next example is a grey level sequence depicting a man walking across a field (Figure 4). There are two difficulties which make it an interesting example for contour tracking. One difficulty is self occlusion. The left leg swings

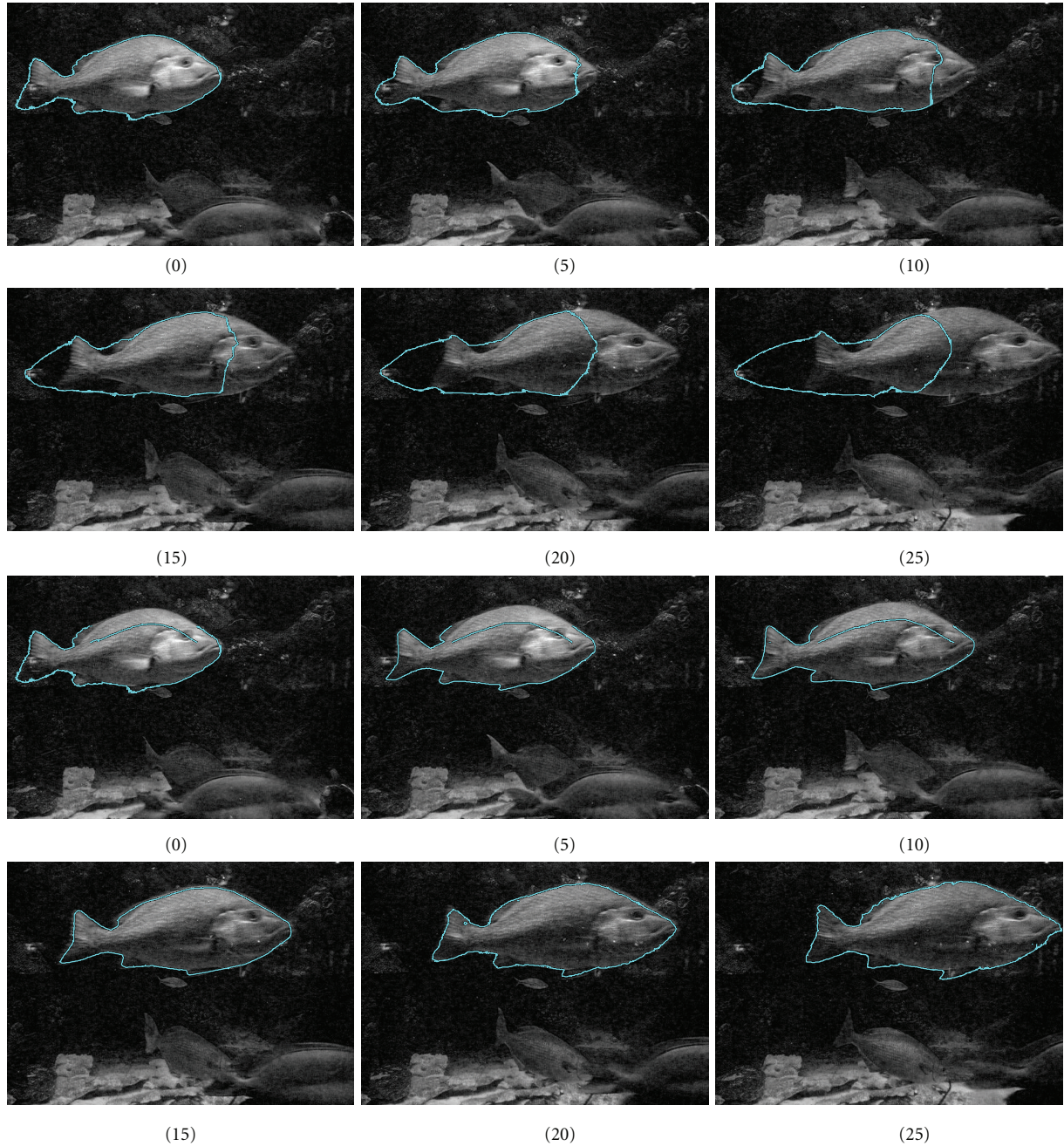


FIGURE 5: Tracking on the *fish* sequence. The first two rows depict results with algorithm in [13]. The last two rows depict tracking results with the proposed method. Note that the shape prior term helped in maintaining a similar tracked shape during evolution. Under each image is the frame number.

forward to progressively occlude the right leg by the 5th frame. The other difficulty is clutter, in the sense that the moving object and the background have a similar random texture along their common border [33]. Clutter of the sort makes matching between frames uncertain and, as a result, the photometric tracking term alone may not sufficiently constrain tracking of the moving boundary. Clutter in this sequence is present, right at the onset, in the left foot/background area, and in the hair/background area. The top two rows of Figure 4 show the results of

applying the algorithm without the shape tracking term in [13]. Without a geometric constraint to prevent it, the active curve encroaches unrestrained on the object region though the blur between the walker feet and the grassy ground. Once in, the shape regularity constraint and the uncertain outcome of intensity matching within the uniform area of the pants keep it drifting inward. The middle row shows tracking results obtained with the method in [10]. Here, the distinction between the intensity histograms inside and outside the curve, on which the evolution equation

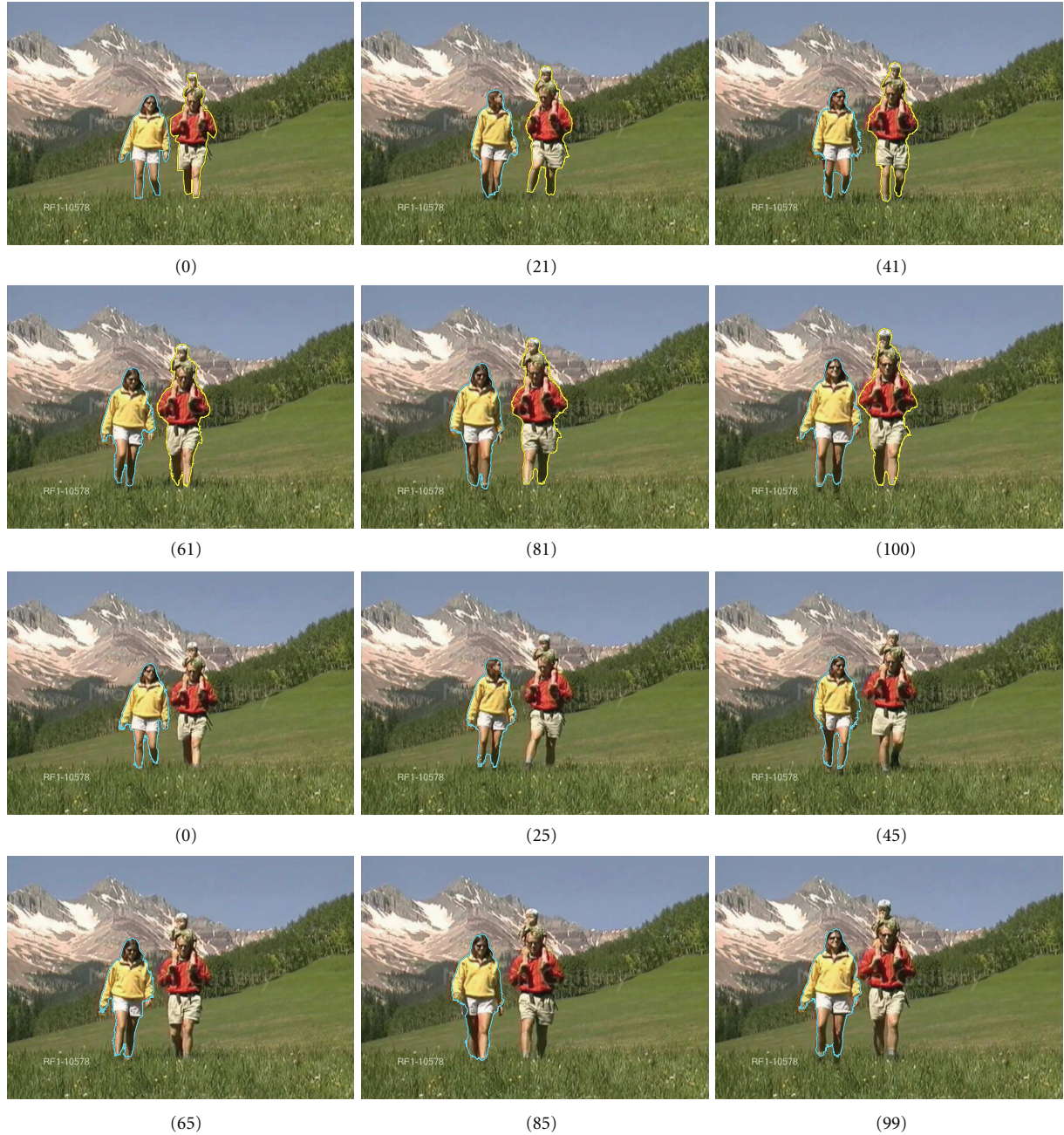


FIGURE 6: Single-object and multiobject tracking. First two rows: two level-set functions are simultaneously evolved. Last two rows: only one level-set function is evolved. Under each image is the frame number.

depends, has only been able to delineate grossly the walker. The bottom two rows depict the results when adding the shape tracking term. The active curve has closely followed the walker silhouette. By constraining the curve to retain a similar shape between consecutive frames, encroachment on the object and subsequent movement inward are averted.

4.3. The Fish Sequence. Fish segmentation and tracking is currently a research problem in a major European study to model fish form and behaviour. The scene in this example (Figure 5) contains several fish swimming in a lighted tank.

The purpose is to track that particular fish in foreground. The main difficulty, peculiar to images of swimming fish, is that every movement of the fish changes the orientation of its mirror-like scales. This causes a variation from frame to frame of the reflected brightness pattern. This variation, which can be significant at times, is difficult to characterize or model. Perusal of the sequence shows that there is also some clutter. It occurs right at the onset of the sequence in the middle portion of the tail. This triangular portion is not covered by skin and the background behind is visible. The initialization includes this part of the background, causing



FIGURE 7: Tracking on the *lifting* sequence with partial occlusion. The tracking method delineates correctly the silhouette of the athlete including parts occluded by the weight bar. The frame index is under each image.

the algorithm correspondence process to look for it in subsequent frames to keep as part of the fish. There is also some clutter in the nose area. Although the initialization delineates the nose correctly, this clutter is a source of confusion as to where the fish/background boundary is in this area. The top two rows in Figure 5 depict the tracking without the shape tracking term. Both difficulties, clutter and the scales orientation variation, have overwhelmed the algorithm. However, the added support of the shape tracking term was able to keep track of the fish boundary throughout the sequence (bottom two rows of Figure 5).

4.4. Multiple-Object Tracking. The purpose of this example (Figure 6) is to illustrate the algorithm's ability to track several objects simultaneously. We mentioned in Section 3.2 that it does this simply by affecting a different level set function to each object contour to track. The level set functions are then evolved independently using evolution equations as in (11). The top two rows show tracking of the woman and man/child independently (blue and yellow

curves). Tracking a single person is shown in the bottom two rows.

4.5. Further Examples. The *lifting* sequence illustrated in Figure 7 is a film of a weight lifting movement. The purpose of the experiment is to use the algorithm to keep track of the outline of the athlete while excluding the weights he is lifting. The athlete's articulated outline changes considerably in the course of the lifting movement. This movement also causes an occlusion of the athlete's body along the weight bar in front. The occluded portion is of relatively small extent in most frames but it covers extensively the arms at about frame 30. However, the presence of distinctive color should be able to lessen the impact of occlusion. This is indeed the case as the active contour adheres to the outline of the athlete's body throughout the sequence.

Another example of sports articulated motion in color sequences is shown in Figure 8. The main challenges in this Karate move images are self-occlusion (the right hand and leg are occluded in most frames) and the significant, fast



FIGURE 8: Tracking on the *Kick* sequence with self occlusion. *Top two rows*: tracking with the proposed method. Although the right hand and leg are often occluded, the method performed well before, during, and after occlusion. *Bottom row*: tracking with the method in [10]: the histogram region intensity representation only allowed a gross delineation of the object.

change in the athlete's outline in the course of the 23-frame sequence. The active curve adheres to the moving boundary throughout the sequence. The bottom row depicts the results obtained using the method in [10]. Similarly to the previous examples, the histogram matching allows keeping track of the moving target grossly without delineating accurately the athlete's silhouette.

One last example is depicted in Figure 9 (walker, color sequence). The right hand is occluded during the first frames. This hand is undetected after it appears in frame 5 because it has no correspondent in previous frames. Of course, it remains undetected in subsequent frames. To handle such cases, a formulation must resort to additional information such as boundary contrast and motion. In this sequence, the object and the background intensity profiles are very distinct; one would not expect the shape term to be influent. Similar results have been obtained by using the method without shape term in [13]. The bottom row shows some of the results obtained using the method in [10]. Even though some parts of the background were included within the tracking contour, the object has been grossly tracked until the end of the sequence.

5. Conclusion

This work investigated a model-free, occlusion accommodating active contour/level set method for tracking moving objects in monocular image sequences. The objective functional contained a photometric tracking term expressed in a kernel-induced space, and a model-free shape tracking term to assist tracking in presence of occlusion and clutter. The method, exclusively driven by the level set evolution equations derived from the objective functional Euler-Lagrange conditions, has performed accurately in several experiments with infrared image sequences, grey level, and color sequences with clutter, partial/total, and self-occlusions. The method can be studied further in several ways. First, a discrete formulation to implement by graph cuts [34] for an execution much faster than level sets can be investigated by inquiry into shape terms consistent with the graph cut formulation. A graph cut statement of this study's method would be very useful because level set implementations can be slow to execute and may not satisfy the requirements of applications such as surveillance. In our examples, the typical run time is 1-2 minutes for video sequences of



FIGURE 9: Tracking of side walk with self occlusion. The contour delineates perfectly the silhouette of the person, except for the appearing right hand clearly visible from frame 5. Bottom row: results by the method in [10].

typically 60 images (we used MATLAB programs run on a Pentium 1.75 GHz processor with 1 Go Memory), and no attempt was made at using special techniques to reduce the computations. Second, new terms can be investigated which bring in motion information without dependence on the availability of accurate motion. Finally, automatic initializations can be investigated so that the method can be of a more general practical use.

Acknowledgment

The authors are grateful to Dr. Tao Zhang and Dr. Daniel Freedman for providing them with their code.

References

- [1] F. Fleuret, J. Berclaz, R. Lengagne, and P. Fua, "Multicamera people tracking with a probabilistic occupancy map," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, no. 2, pp. 267–282, 2008.
- [2] D. A. Migliore, M. Matteucci, and M. Naccari, "A reevaluation of frame difference in fast and robust motion detection," in *Proceedings of the 4th ACM International Workshop on Video Surveillance and Sensor Networks (VSSN '06)*, pp. 215–218, ACM, New York, NY, USA, October 2007.
- [3] C. J. Veenman, M. J. T. Reinders, and E. Backer, "Resolving motion correspondence for densely moving points," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23, no. 1, pp. 54–72, 2001.

- [4] D. Comaniciu, V. Ramesh, and P. Meer, "Kernel-based object tracking," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, no. 5, pp. 564–577, 2003.
- [5] M. E. Leventon, W. E. L. Grimson, and O. Faugeras, "Statistical shape influence in geodesic active contours," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR '00)*, vol. 1, pp. 316–323, June 2000.
- [6] D. Cremers and S. Soatto, "A pseudo distance for shape priors in level set segmentation," in *Proceedings of the 2nd IEEE Workshop on Variational, Geometric and Level Set Methods in Computer Vision*, N. Paragios, Ed., pp. 169–176, October, 2003.
- [7] X. Bresson, P. Vandergheynst, and J. P. Thiran, "A variational model for object segmentation using boundary information and shape prior driven by the Mumford-Shah functional," *International Journal of Computer Vision*, vol. 68, no. 2, pp. 145–162, 2006.
- [8] A. Mitiche and I. Ben Ayed, *Variational and Level Set Methods in Image Segmentation*, Springer, 1st edition, 2010.
- [9] S. Jehan-Besson, M. Barlaud, and G. Aubert, "Detection and tracking of moving objects using a new level set based method," in *Proceedings of the International Conference on Pattern Recognition*, pp. 7112–7117, Barcelona, Spain, September 2000.
- [10] D. Freedman and T. Zhang, "Active contours for tracking distributions," *IEEE Transactions on Image Processing*, vol. 13, no. 4, pp. 518–526, 2004.
- [11] D. Cremers, "Dynamical statistical shape priors for level set-based tracking," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, no. 8, pp. 1262–1273, 2006.
- [12] A. Yilmaz, X. Li, and M. Shah, "Contour-based object tracking with occlusion handling in video acquired using mobile cameras," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, no. 11, pp. 1531–1536, 2004.
- [13] A. R. Mansouri, "Region tracking via level set PDEs without motion computation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 7, pp. 947–961, 2002.
- [14] I. Ben Ayed, S. Li, and I. Ross, "Embedding overlap priors in variational left ventricle tracking," *IEEE Transactions on Medical Imaging*, vol. 28, no. 12, pp. 1902–1913, 2009.
- [15] T. Chan and W. Zhu, "Level set based shape prior segmentation," in *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '05)*, vol. 2, pp. 1164–1170, June 2005.
- [16] A. R. Mansouri and A. Mitiche, "Region tracking via local statistics and level set PDEs," in *Proceedings of International Conference on Image Processing (ICIP '02)*, vol. 3, pp. 605–608, Rochester, NY, USA, June 2002.
- [17] S. Chen and R. J. Radke, "Level set segmentation with both shape and intensity priors," in *Proceedings of the 12th International Conference on Computer Vision (ICCV '09)*, pp. 763–770, October 2009.
- [18] P. Bouboulis, K. Slavakis, and S. Theodoridis, "Adaptive kernel-based image denoising employing semi-parametric regularization," *IEEE Transactions on Image Processing*, vol. 19, no. 6, pp. 1465–1479, 2010.
- [19] M. Ben Salah, A. Mitiche, and I. Ben Ayed, "Effective level set image segmentation with a kernel induced data term," *IEEE Transactions on Image Processing*, vol. 19, no. 1, pp. 220–232, 2010.
- [20] M. Ben Salah, A. Mitiche, and I. Ben Ayed, "Multiregion image segmentation by parametric kernel graph cuts," *IEEE Transactions on Image Processing*, vol. 20, no. 2, pp. 545–557, 2011.
- [21] S. Geman and D. Geman, "Stochastic relaxation, gibbs distributions, and the bayesian restoration of images," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 6, no. 6, pp. 721–741, 1984.
- [22] K.-R. Müller, S. Mika, G. Ratsch, S. Tsuda, and B. Schölkopf, "An introduction to kernel-based learning algorithms," *IEEE Transactions on Neural Networks*, vol. 12, no. 2, pp. 181–202, 2001.
- [23] B. Schölkopf, A. Smola, and K.-R. Müller, "Nonlinear component analysis as a kernel eigenvalue problem," *Neural Computation*, vol. 10, no. 5, pp. 1299–1319, 1998.
- [24] I. S. Dhillon, Y. Guan, and B. Kulis, "Weighted graph cuts without eigenvectors a multilevel approach," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 11, pp. 1944–1957, 2007.
- [25] P. J. Huber, *Robust Statistics*, John Wiley & Sons, New York, NY, USA, 1981.
- [26] J. A. Sethian, *Level Set Methods and Fast Marching Methods*, Cambridge University Press, 1999.
- [27] D. Cremers, M. Rousson, and R. Deriche, "A review of statistical approaches to level set segmentation: integrating color, texture, motion and shape," *International Journal of Computer Vision*, vol. 72, no. 2, pp. 195–215, 2007.
- [28] R. Malladi, J. A. Sethian, and B. C. Vemuri, "Shape modeling with front propagation: a level set approach," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 17, no. 2, pp. 158–175, 1995.
- [29] V. Caselles, R. Kimmel, and G. Sapiro, "Geodesic active contours," *International Journal of Computer Vision*, vol. 22, no. 1, pp. 61–79, 1997.
- [30] D. Cremers, S. J. Osher, and S. Soatto, "Kernel density estimation and intrinsic alignment for shape priors in level set segmentation," *International Journal of Computer Vision*, vol. 69, no. 3, pp. 335–351, 2006.
- [31] T. F. Chan and L. A. Vese, "Active contours without edges," *IEEE Transactions on Image Processing*, vol. 10, no. 2, pp. 266–277, 2001.
- [32] L. Alparone, G. Corsini, and M. Diani, "Noise modeling and estimation in image sequences from thermal infrared cameras," in *Image and Signal Processing for Remote Sensing X*, vol. 5573 of *Proceedings of SPIE*, pp. 381–389, September 2004.
- [33] G. Waldman, J. Wootton, G. Hobson, and K. Luetkemeyer, "A normalized clutter measure for images," *Computer Vision, Graphics, & Image Processing*, vol. 42, no. 2, pp. 137–156, 1988.
- [34] Y. Boykov and V. Kolmogorov, "An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, no. 9, pp. 1124–1137, 2004.

