Tracking Deforming Objects using Particle Filtering for Geometric Active Contours

Yogesh Rathi, Namrata Vaswani, Allen Tannenbaum, Anthony Yezzi

Abstract

Tracking deforming objects involves estimating the global motion of the object and its local deformations as a function of time. Tracking algorithms using Kalman filters or Particle filters have been proposed for finite dimensional representations of shape, but these are dependent on the chosen parametrization and cannot handle changes in curve topology. Geometric active contours provide a framework which is parametrization independent and allow for changes in topology. In the present work, we formulate a particle filtering algorithm in the geometric active contour framework that can be used for tracking moving and deforming objects.

I. Introduction

The problem of tracking moving and deforming objects has been a topic of substantial research in the field of controlled active vision; see [1], [2] and the references therein. In this paper, we propose a scheme which combines the advantages of particle filtering and geometric active contours realized via level set models for dynamic tracking.

In order to appreciate this methodology, we briefly review some previous related work. First of all, a number of different representations of shape have been proposed in literature together with algorithms for tracking using such representations. In particular, the notion of *shape* has been found to be very useful in this enterprize. For example, the shape of a set of N discrete points (called *landmarks*) in $\mathbf{R}^M$ is defined as the equivalence class of $\mathbf{R}^{MN}$ under the Euclidean similarity group in $\mathbf{R}^M$. The dynamics of the similarity group defines the global motion while the dynamics of the equivalence class defines the deformation. In [3], the authors define a prior

Yogesh Rathi, Namrata Vaswani, Allen Tannenbaum, Anthony Yezzi are with the School of Electrical & Computer, Georgia Institute of Technology, Atlanta, GA 30332-0250; {yogesh.rathi, tannenba}@bme.gatech.edu,{namrata, ayezzi}@ece.gatech.edu.
This research was supported by grants from NSF, NIH, AFOSR, ARO, and ONR.

dynamical model on the deformation and on the similarity group parameters. A particle filter [4] is then used to track the deformation and the global motion over time.

The possible parameterizations of shape are of course very important. We should note that various finite dimensional parameterizations of continuous curves have been proposed, perhaps most prominently the B-spline representation used for a "snake model" as in [2]. Isard and Blake (see [1] and references therein) apply the B-spline representation for contours of objects and propose the Condensation algorithm [5] which treats the affine group parameters as the state vector, learns a prior dynamical model for them and uses a particle filter [4] to estimate them from the noisy observations. Since this approach only tracks the affine parameters it cannot handle local deformations of the deforming object (see e.g., the fish example in Section IV-A).

Another approach for representing contours is via the level set technique [6], [7] which is an implicit representation of contours. For segmenting a shape using level sets, an initial guess of the contour is deformed until it minimizes an image-based energy functional. Some previous work on tracking using level set methods is given in [8], [9], [10], [11].

The work in this paper extends the ideas presented in [10], [11]. More precisely, in [10], the authors propose a definition for motion and shape deformation for a deforming object. Motion is parameterized by a finite dimensional group action (e.g. Euclidean or Affine) while shape deformation is the total deformation of the object contour (infinite dimensional group) modulo the finite dimensional motion group. This is called *deformotion*. Tracking is then defined as a trajectory on the finite dimensional motion group. This approach relies only on the observed images for tracking and does not use any prior information on the dynamics of the group action or of the deformation. As a result it fails if there is an outlier observation or if there is occlusion. To address this problem, [11] proposes a generic local observer to incorporate prior information about the system dynamics in the "deformotion" framework. They impose a constant velocity prior on the group action and a zero velocity prior on the contour. The observed value of the group action and the contour is obtained by a joint minimization of the energy. This is linearly combined with the value predicted by the system dynamics using an observer matrix.

This approach suffers from two problems. First, as in [10], they must perform a joint minimization over the group action and the contour at each time step which is computationally very intensive. Second, for nonlinear systems such as the one used in [11], there is no systematic way to choose the observer matrix to guarantee stability. The present paper addresses the above limitations. We formalize the incorporation of a prior system model along with an observation model. A particle filter is used to estimate the conditional probability distribution of the group action and the contour at time t, conditioned on all observations up to time t.

Other approaches closely related to our work are given in [2], [12]. Here the authors use a Kalman filter in conjunction with active contours to track nonrigid objects. The Kalman filter was used for predicting possible movements of the object, while the active contours allowed for tracking deformations in the object. The literature discussed above is by no means exhaustive. Due to paucity of space, we have discussed only a few related works here.

This note is organized as follows: In Section 2 we describe the state space model and Section 3 discusses the algorithm in detail. Experimental results are given in Section 4. Limitations and future work are discussed in Section 5.

II. The System and Observation Model and Importance Sampling

Let C_t denote the contour at time t. The basic idea of the level set approach is to embed the contour C_t as the zero level set of a graph of a higher dimensional function $\Phi : \mathbf{R^2} \longrightarrow \mathbf{R}$ and then evolve the graph so that this level set moves according to the given curve evolution equation. Level sets have the advantage of being parameter independent (i.e. they are implicit representation of the curve) and can handle topological changes naturally. The particle filter [4], [13] is a sequential Monte Carlo method which produces at each time t, a cloud of N particles, $\{X_t^{(i)}\}_{i=1}^N$, whose empirical measure closely "follows" $p(X_t|Y_{1:t})$, the posterior distribution of the state given past observations. It was first introduced in [4] as the Bayesian Bootstrap filter and its first application to tracking in computer vision was the Condensation algorithm [5].

Let A_t denote a 6-dimensional affine parameter vector with the first 4 parameters representing rotation, skew and scale and the last 2 parameters representing translation. We propose to use the affine parameters (A_t) and the contour (C_t) as the state, i.e. $X_t = [A_t, C_t]$ and treat the image at time t as the observation, i.e. $Y_t = \text{Image}(t)$. The prediction step for X_t consists of predicting the affine motion of the object followed by predicting the deformation. The affine motion prediction is done by using a first or second order (constant velocity or acceleration) autoregressive (AR) model on the affine parameters. So we have,

$$A_t = f_{AR}(A_{t-1}, u_t), \quad \mu_t = A_t(C_{t-1}) \ , \tag{1}$$

where u_t is a 6-dim Gaussian noise vector and f_{AR} is the AR model explained in Section III-A.

When the image at time t (Y_t) is available, μ_t is deformed by doing gradient descent (commonly referred to as "curve evolution") on the image energy E_{image} (any image dependent energy functional, see Section III-C) at time t, followed by addition of random Gaussian noise. Thus we have

$$C_t = f^L_{CE}(\mu_t, Y_t) + u_{C,t} \tag{2}$$

where $u_{C,t}$ is an n-dimensional[1] noise vector with distribution $\mathcal{N}(0, \Sigma)$ and $f^L_{CE}(\mu, Y)$ is given by L iterations of gradient descent. Doing curve evolution accounts for using the latest observation Y_t to obtain local shape deformation and position of the object. This step can be interpreted as importance sampling from a proposal distribution dependent on the current observation (discussed below).

Now $f^L_{CE}(\mu, Y)$ is given by

$$\mu^k = \mu^{k-1} - \alpha \nabla_\mu E_{image}(\mu^{k-1}, Y),\ k = 1, 2, 3, .., L$$

$$\text{where} \quad \mu^0 = \mu \quad \text{and} \quad f^L_{CE}(\mu, Y) = \mu^L \ .$$

Note that we fixed $L = 4$ in our experiments. If μ_t is evolved until convergence, one would reach a local minimum of the energy E_{image}. This is not desirable since the local minimum would be independent of all starting contours in its domain of attraction and would only depend on the observation, Y_t. Thus the state at time t would loose its dependence on the state at time $t - 1$ and this may cause loss of track in cases where the observation is bad. But if μ_t is evolved only a fixed number of times, it will deviate the contour only a little (in a direction which decreases the energy E_{image} as fast as possible using only local information) so that particles are moved to regions of high likelihood.

The "likelihood" i.e., probability of observation $Y_t = \text{Image}(t)$ given state X_t, is given by:

$$p(Y_t|X_t) \propto e^{-E_{image}(Y_t, C_t)} \ . \tag{3}$$

We now explain *importance sampling* [13] and how we use it in our particle filtering algorithm (described in the next section). Suppose $p(x)$ is a probability density from which it is difficult to draw samples and $q(x)$ is a density which is easy to sample from and has a heavier tail than $p(x)$

[1] n is the number of points representing the deformed contour $f^L_{CE}(\mu_t, Y_t)$ on a discrete grid and it varies with time.

(i.e. there exists a bounded region R such that for all points outside R, $q(x) > p(x)$). $q(x)$ is known as the *proposal density* or the *importance density*. Let $x^i \sim q(x)$, $i = 1, ..., N$ be samples generated from $q(.)$. Then, an approximation to $p(.)$ is given by $p(x) \approx \sum_{i=1}^{N} w^i \delta(x - x^i)$, where $w^i \propto \frac{p(x^i)}{q(x^i)}$ is the normalized weight of the i-th particle. So, if the samples, $X_t^{(i)}$, were drawn from an importance density, $q(X_t|X_{1:t-1}, Y_{1:t})$, and weighted by $w_t^{(i)} \propto \frac{p(X_t^{(i)}|Y_{1:t})}{q(X_t^{(i)}|X_{1:t-1}^{(i)}, Y_{1:t})}$, then $\sum_{i=1}^{N} w_t^{(i)} \delta(X_t^{(i)} - X_t)$ approximates $p(X_t|Y_{1:t})$.

In our case, the state process is a Markov process and $p(Y_t|X_{0:t}, Y_{0:t-1}) = p(Y_t|X_t)$ (sometimes referred to as the "memoryless channel assumption") and since we take the importance sampling density $q(X_t|X_{0:t-1}, Y_{1:t}) = q(X_t|X_{t-1}, Y_t)$, we get the following recursion for the weights [13]:

$$w_t^{(i)} \propto w_{t-1}^{(i)} \frac{p(Y_t|X_t^{(i)}) p(X_t^{(i)}|X_{t-1}^{(i)})}{q(X_t^{(i)}|X_{t-1}^{(i)}, Y_t)} \ . \tag{4}$$

The importance density can be written as[2]

$$q(X_t|X_{t-1}, Y_t) = q(A_t, C_t|A_{t-1}, C_{t-1}, Y_t) = q(A_t|A_{t-1}) q(C_t|A_t(C_{t-1}), Y_t) = p(A_t|A_{t-1}) q(C_t|\mu_t, Y_t),$$

since we sample A_t from $p(A_t|A_{t-1})$, we have $q(A_t|A_{t-1}) = p(A_t|A_{t-1})$.

The prior density $p(X_t|X_{t-1})$ can be written as:

$$p(X_t|X_{t-1}) = p(A_t, C_t|A_{t-1}, C_{t-1}) = p(A_t|A_{t-1}) p(C_t|A_t(C_{t-1})) = p(A_t|A_{t-1}) p(C_t|\mu_t)$$

Thus, (4) can be written as:

$$w_t^{(i)} \propto w_{t-1}^{(i)} \ \frac{p(Y_t|X_t^{(i)}) \ p(C_t^{(i)}|\mu_t^{(i)})}{q(C_t^{(i)}|\mu_t^{(i)}, Y_t)} \ . \tag{5}$$

The probability $p(C_t|\mu_t)$ can be calculated using any suitable measure of similarity between shapes. One such measure is to take $p(C_t|\mu_t) \propto e^{-d^2(C_t, \mu_t)}$ where d^2 is the dissimilarity measure given by equation (14) in Section III-E.

The choice of the importance density is a critical design issue for implementing a successful particle filter. As described in [14], the proposal distribution q should be such that particles generated by it lie in the regions of high observation likelihood. One way of doing this is to use a proposal density which depends on the current observation. This idea has been used in many past works such as the unscented particle filter [14] where the proposal density is a Gaussian

[2] Note that the curve obtained after doing curve evolution is denoted by C_t, while the curve obtained by applying the affine transformation is denoted by μ_t, i.e., $\mu_t = A_t(C_{t-1})$.

density with a mean that depends on the current observation. Our update step described in equation (2) can be interpreted as importance sampling from the density $q(C_t|\mu_t, Y_t)$ given by $\mathcal{N}(f_{CE}(\mu_t, Y_t), \Sigma)$ where Σ is an $n \times n$ (n is the number of points representing the contour C_t on a discrete grid and hence varies with time) covariance matrix. The covariance of the noise should be large enough so that q has a heavier tail than $p(Y_t|X_t)p(X_t|X_{t-1})$ (in the sense defined above). Note that, in practice it is not possible to evaluate Σ satisfying this condition.

III. The Particle Filtering Algorithm

Based on the description above, the proposed algorithm can be written as follows:

1) **Importance Sampling:**
   a) Generate samples $\{A_t^{(i)}, \mu_t^{(i)}\}_{i=1}^N$ using:

$$A_t^{(i)} = f_{AR}(A_{t-1}^{(i)}, u_t^{(i)}), \quad \mu_t{}^{(i)} = A_t^{(i)}(C_{t-1}^{(i)}) \ .$$

   b) Perform L steps of curve evolution on each $\mu_t^{(i)}$ and add noise:

$$C_t^{(i)} = f_{CE}^L(\mu_t^{(i)}, Y_t) + u_{C,t}^{(i)}, \quad u_{C,t}^{(i)} \sim \mathcal{N}(\mathbf{0}, \Sigma) \ . \tag{6}$$

2) **Weighting and Resampling:**
   a) Calculate weights and normalize:

$$\tilde{w}_t^{(i)} = \frac{e^{-E_{image}(Y_t, C_t^{(i)})} \ e^{-d^2(C_t^{(i)}, \mu_t^{(i)})}}{q(C_t^{(i)}|\mu_t^{(i)}, Y_t)}, \ w_t^{(i)} = \frac{\tilde{w}_t^{(i)}}{\sum_{j=1}^N \tilde{w}_t^{(j)}} \ .$$

   where d is defined in (14) in Section III-E and E_{image} is defined in Section III-C.
   b) Resample to generate N particles $\{A_t^{(i)}, C_t^{(i)}\}$ distributed according to $p(A_t, C_t|Y_{1:t})$,

$$p(A_t, C_t|Y_{1:t}) \approx \sum_{i=1}^N \frac{1}{N} \delta_{A_t^{(i)}, C_t^{(i)}}(A_t, C_t) \ ,$$

3) Go back to the importance sampling step for $t+1$.

The resampling step improves sampling efficiency by eliminating particles with very low weights. Other details of the above algorithm are discussed in the following subsections.

A. The AR model

In the above algorithm f_{AR} could be any suitable prediction function which can model the dynamics of motion of the moving object. Rather than conjuring up a model that is merely plausible, one can learn the dynamics of motion from a training set. This can be done using an autoregressive (AR) model. A second-order AR process in which the affine parameters at a given time depend on two previous time-steps is given by:

$$A_{t+1} - \bar{A} = B_1(A_t - \bar{A}) + B_2(A_{t-1} - \bar{A}) + B_0 u_{t+1} \ , \tag{7}$$

where A_t is the 6-dimensional affine parameter vector (10), B_1, B_2, B_0 are 6×6 matrices learned *a priori*, u_{t+1} is a vector of 6 independent random $\mathcal{N}(0,1)$ variables and $\bar{A}$ is the steady state mean of the model. We refer the interested reader to [1] for further details on how to learn these parameter matrices and the advantages of using the second-order model (AR-2) versus the first-order model (AR-1).

B. Learning Affine Motion

Many approaches [15] have been reported in the literature for finding the affine parameters that relate one image to the other. Most of these methods require a set of feature points to be known before one can find the affine parameters that relate them. In [16] the author proposes a method to find the affine parameters using only the source and target images. The affine transformation that relates the curve $C(t)$ and $C(t-1)$ is given by:

$$C(x, y, t) = C(m_1 x + m_2 y + m_5, m_3 x + m_4 y + m_6, t-1) \ ,$$

where, m_i are the affine parameters. In order to estimate these parameters, the following quadratic error is to be minimized:

$$E(\vec{m}) \ = \ \sum_{x,y \in \omega} [C(x, y, t) - C(m_1 x + m_2 y + m_5, m_3 x + m_4 y + m_6, t-1)]^2 \ ,$$

which is linearized and then minimized to give

$$\vec{m} = \left[\sum_{x,y \in \omega} \vec{d}\, \vec{d}^T \right]^{-1} \left[\sum_{x,y \in \omega} \vec{d}\, k \right] \ , \tag{8}$$

where the scalar k and the vectors $\vec{d}$, $\vec{m}$ are given as[3]:

$$k = C_t + xC_x + yC_y \quad \text{and} \quad \vec{d}^T = (xC_x \quad yC_x \quad xC_y \quad yC_y \quad C_x \quad C_y) \tag{9}$$

$$\vec{m} = (m_1 \quad m_2 \quad m_3 \quad m_4 \quad m_5 \quad m_6)^T \ . \tag{10}$$

Derivation details are available in [16]. Once the affine parameter vector $\vec{m}$ is known for the training set, the AR model parameter matrices can be learned as given in [1].

C. The Model of Chan and Vese

Many methods [8], [17], [18] which incorporate geometric and/or photometric (color, texture, intensity) information have been shown to segment images robustly in presence of noise and clutter. In the prediction step above, f_{CE} could be any edge based or region based (or a combination of both) curve evolution equation. In our numerical experiments we have used the Mumford-Shah functional [19] as modelled by Chan and Vese [20] to obtain the curve evolution equation, which we describe briefly. We seek to minimize the following energy:

$$\begin{aligned} E_{image} = E_{cv}(c_1, c_2, \Phi) = \int_\Omega (f - c_1)^2 H(\Phi) dx\ dy + \int_\Omega (f - c_2)^2 (1 - H(\Phi))\ dx\ dy \\ + \nu \int_\Omega |\nabla H(\Phi)| dx\ dy \ , \end{aligned} \tag{11}$$

where c_1 and c_2 are defined as:

$$c_1 = \frac{\int f(x,y) H(\Phi) dx\ dy}{\int H(\Phi) dx\ dy}, c_2 = \frac{\int f(x,y)(1 - H(\Phi)) dx\ dy}{\int (1 - H(\Phi)) dx\ dy} \ ,$$

and $H(\Phi)$ is the Heaviside function defined as:

$$H(\Phi) = \begin{cases} 1 & \Phi \geq 0 \ , \\ 0 & else \ , \end{cases} \tag{12}$$

and finally $f(x, y)$ is the image and Φ is the level set function as defined in Section II before. The Euler-Lagrange equation for this functional can be implemented by the following gradient descent [20], [19]:

$$\frac{\partial \Phi}{\partial t} = \delta_\epsilon(\Phi) \left[\nu \ \text{div} \left(\frac{\nabla \Phi}{|\nabla \Phi|} \right) - (f - c_1)^2 + (f - c_2)^2 \right] \quad \text{where} \quad \delta_\epsilon(s) = \frac{\epsilon}{\pi(\epsilon^2 + s^2)} \ .$$

[3]The subscripts in this equation denote partial derivatives.

D. Dealing with Multiple Objects

In principle, the Condensation filter [1] could be used for tracking multiple objects. The posterior distribution will be multi-modal with each mode corresponding to one object. However, in practice it is very likely that a peak corresponding to the dominant likelihood value will increasingly dominate over all other peaks when the estimation progresses over time. In other words, a dominant peak is established if some objects obtain larger likelihood values more frequently. So, if the posterior is propagated with fixed number of samples, eventually, all samples will be around the dominant peak. This problem becomes more pronounced in cases where the objects being tracked do not have similar photometric or geometric properties. We deal with this issue as given in [21] by first finding the clusters within the state density to construct a Voronoi tessalation [22] and then resampling within each Voronoi cell separately as follows: 1) Every step, build an importance function which results in equal number of samples being taken in each Voronoi cell. 2) Every N steps, rescale the weights in each cell so that the peak weight is 1.

Other solutions proposed by [23], [24] could also be used in tackling this problem of sample impoverishment.

E. Coping with Occlusions

Many active contour models [18], [17], [25] which use shape information have been reported in the literature. Prior shape knowledge is necessary when dealing with occlusions. In particular, in [8], the authors incorporate "shape energy" in the curve evolution equation to deal with occlusions. Any such energy term can be used in the proposed model to deal with occlusions. In numerical experiments we have dealt with this issue in a slightly different way by incorporating the shape information in the weighting step instead of the curve evolution step, i.e. we calculate the likelihood probability for each particle using the following:

$$p(Y_t|X_t^{(i)}) = \lambda_1 \left(\frac{e^{-E_{cv}^{(i)}}}{\sum_{j=1}^{N} e^{-E_{cv}^{(j)}}}\right) + \lambda_2 \left(1 - \frac{d^2(\Phi^{(s)}, \Phi^{(i)})}{\sum_{j=1}^{N} d^2(\Phi^{(s)}, \Phi^{(j)})}\right) , \tag{13}$$

where $\lambda_1 + \lambda_2 = 1$ and $d^2(\Phi^{(s)}, \Phi^{(i)})$ is the dissimilarity measure as given in [25] by,

$$d^2(\Phi^{(s)}, \Phi^{(i)}) = \int_\Omega (\Phi^{(s)} - \Phi^{(i)})^2 \, \frac{h(\Phi^{(s)}) + h(\Phi^{(i)})}{2} \, dx \, dy \quad \text{with} \quad h(\Phi) = \frac{H(\Phi)}{\int_\Omega H(\Phi) \, dx \, dy} \tag{14}$$

where $\Phi^{(s)}$ and $\Phi^{(i)}$ are the level set functions of a template shape and the i-th contour shape respectively and $H(\Phi)$ is the Heaviside function as defined before in (12). The dissimilarity measure gives an estimate of how different any two given shapes (in particular, their corresponding level sets) are. So, higher values of d^2 indicates more dissimilarity in shape. Using this strategy, particles which are closer to the template shape are more likely to be chosen than particles with "occluded shapes" (i.e., shapes which include the occlusion).

IV. Experiments

In this section we describe some experiments performed to test the proposed tracking algorithm. We certainly do not claim that the method proposed in this note is optimal, but only claim that to the best of our knowledge this is the first time geometric active contours in a level set framework have been used in conjunction with the particle filter [4] for tracking deforming objects. Results of applying the proposed method on three image sequences are given below. The model of Chan and Vese [20], as described earlier, was used for curve evolution. Level set implementation was done using narrow band evolution [7]. Learning [1] was performed on images without the background clutter, i.e. on the outlines of the object. In numerical experiments, there was no noticeable difference between results obtained by adding noise to the contour C_t (see equation (6)) versus those obtained without adding noise. The results shown in this paper were obtained without adding noise to the contour.

A. Fish Sequence

In the fish video, the shape of the fish undergoes sudden deformation as the fish turns or gets partially occluded (see Figure 3, Frames 167, 181). This local shape deformation cannot be modelled using an affine motion model. Hence, such motion is difficult to track using the standard Condensation filter [1]. As can been seen in the images, (Figure 3) the proposed method can robustly track nonrigid deformations in the shape of the fish. Note that, *no* shape information either in curve evolution or in the weighting step was used in tracking this sequence, i.e. we did not use the dissimilarity term specified in Section III-E. For this test sequence, an AR-1 model [1] was used for affine motion prediction.

B. Car Sequence

In this sequence, the car is occluded as it passes through the lamp post. Trying to track such a sequence using geometric active contours (for example, (13)) without any "shape energy" gives

(a) Frame 35 (b) Frame 47 (c) Frame 59

Fig. 1. Tracking using equation (13) without particle filter

very poor results as shown in Figure 1. However, using the proposed method and a weighting strategy as described in Section III-E the car can be successfully tracked (Figure 2). Note that we used equation (13) for the curve evolution which does not contain any shape term. A second-order autoregressive model (7) was used for f_{AR}.

C. Couple Sequence

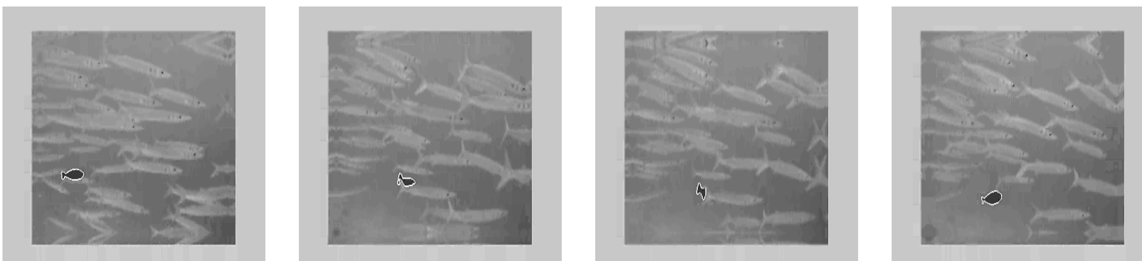
The walking couple sequence demonstrates multiple object tracking. In general, tracking such a sequence by the standard Condensation method [1] can give erroneous results when the couple come very close to each other or touch each other, since the measurements made for the person on the right can be interpreted by the algorithm as coming from the left. One solution has been proposed in [23]. Our method naturally avoids this problem since it uses "region based" energy E_{cv} (11) and weighting as given in Section III-E to find the observation probabilities. To track multiple objects, we used the method described in Section III-D. Since the number of frames in the video is less (about 22) no dynamical motion model was learnt, resulting in the state transition equation: $A_t = A_{t-1} + Bu_t$ where u_t is white Gaussian noise and B is a known covariance matrix which is assumed to be constant through the state evolution process. This video demonstrates the fact that, the proposed algorithm can track robustly (see Figure 4) even when the learnt model is completely absent.

V. Limitations and Future Work

In this paper, we proposed a particle filtering algorithm for geometric active contours which can be used for tracking moving and deforming objects. The proposed method can deal with partial occlusions and can track robustly even in the absence of a learnt model.

(a) Frame 16 (b) Frame 39 (c) Frame 48 (d) Frame 64

Fig. 2. Car Sequence



(a) Frame 34 (b) Frame 167 (c) Frame 181 (d) Frame 215

Fig. 3. Fish Sequence

The above framework has several limitations which we intend to overcome in our future work. First, we have to include some kind of shape information when we track objects which undergo major occlusions. This restricts our ability to track highly deformable objects in such situations. Secondly, the algorithm might perform poorly if the object being tracked is *completely occluded* for many frames. In [26], the authors have performed PCA on a set of signed distance functions of training shapes to obtain principal directions of variation of the signed distance function. We can adopt a similar idea and add noise in the principal variation directions to obtain contour prediction. This approach can also provide a shape prior.

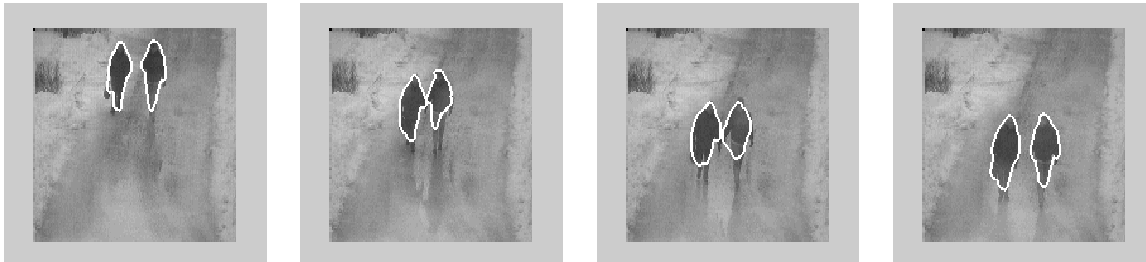
VI. Acknowledgements

We thank Marc Niethammer and Oleg Michailovich for fruitful discussions.

References

[1] A. Blake and M. Isard, Eds., *Active Contours*. Springer, 1998.

[2] D. Terzopoulos and R. Szeliski, *Active Vision*. MIT Press, 1992, ch. Tracking with Kalman Snakes, pp. 3–20.



(a) Frame 2 (b) Frame 9 (c) Frame 15 (d) Frame 18

Fig. 4. Couple Sequence

[3] N. Vaswani, A. RoyChowdhury, and R. Chellappa, "Activity recognition using the dynamics of the configuration of interacting objects," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Madison, WI, June 2003.

[4] N. Gordon, D. Salmond, and A. Smith, "Novel approach to nonlinear/nongaussian bayesian state estimation," *IEE Proceedings-F (Radar and Signal Processing)*, pp. 140(2):107–113, 1993.

[5] M. Isard and A. Blake, "Condensation – conditional density propagation for visual tracking," *International Journal of Computer Vision*, vol. 29, no. 1, pp. 5–28, 1998.

[6] S. J. Osher and J. A. Sethian, "Fronts propagation with curvature dependent speed: Algorithms based on hamilton-jacobi formulations," *Journal of Computational Physics*, vol. 79, pp. 12–49, 1988.

[7] J. A. Sethian, *Level Set Methods and Fast Marching Methods*, 2nd ed. Cambridge University Press, 1999.

[8] T. Zhang and D. Freedman, "Tracking objects using density matching and shape priors," in *Proceedings of the Ninth International Conference on Computer Vision*. IEEE, 2003, pp. 1950–1954.

[9] N. Paragois and R. Deriche, "Geodesic active contorus and level sets for the detection and tracking of moving objects," *Transactions on Pattern analysis and Machine Intelligence*, vol. 22, no. 3, pp. 266–280, 2000.

[10] A. Yezzi and S. Soatto, "Deformotion: Deforming motion, shape average and the joint registration and approximation of structures in images," *Internaitonal Journal of Computer Vision*, vol. 53, no. 2, pp. 153–167, 2003.

[11] J. Jackson, A. Yezzi, and S. Soatto, "Tracking deformable moving objects under severe occlusions," in *Conf. decision and control*, Dec, 2004.

[12] N. Peterfreund, "Robust tracking of position and velocity with Kalman snakes," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 21, no. 6, pp. 564–569, 1999.

[13] A. Doucet, N. deFreitas, and N. Gordon, *Sequential Monte Carlo Methods in Practice*. Springer, 2001.

[14] R. van der Merwe, N. de Freitas, A. Doucet, and E. Wan, "The unscented particel filter," in *Advances in Neural Information Processing Systems 13*, Nov 2001. [Online]. Available: citeseer.ist.psu.edu/article/vandermerwe00unscented.html

[15] Z. Yand and F. Cohen, "Cross-weighted moments and affine invariants for image registration and matching," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 21, no. 8, 1999.

[16] S. Periaswamy and H. Farid, "Elastic registration in the presence of intensity variations," *IEEE Transactions on Medical Imaging*, 2003.

[17] M. Rousson and N. Paragois, "Shape priors for level set representations," in *Proceedings of European Conference on Computer Vision*, 2002, pp. 78–92.

[18] M. Leventon, W. L. Grimson, and O. Faugeras, “Statistical shape influence in geodesic active contours,” in *Proc. CVPR*. IEEE, 2000, pp. 1316–1324.

[19] D. Mumford and J. Shah, “Optimal approximation by piecewise smooth functions and associated variational problems,” *Commun. Pure Applied Mathematics*, vol. 42, pp. 577–685, 1989.

[20] T. Chan and L. Vese, “Active contours without edges,” *IEEE Trans. on Image Processing*, vol. 10, pp. 266–277, 2001.

[21] D. Tweed and A. Calway, “Tracking many objects using subordinated condensation,” in *The British Machine Vision Conference*, 2002, pp. 283–292.

[22] R. Sedgewick, *Algorithms*. Addison-Wesley, 1992.

[23] H. Tao, H. Sawhney, and R. Kumar, “A sampling algorithm for tracking multiple objects,” in *Proceedings of Vision Algorithms, ICCV*, 1999.

[24] J. MacCormick and A. Blake, “A probabilistic exclusion principle for tracking multiple objects,” *International Journal of Computer Vision*, vol. 39, pp. 57–71, 2000.

[25] D. Cremers and S. Soatto, “A pseudo-distance for shape priors in level set segmentation,” in *IEEE Workshop on Variational, Geometric and Level Set Methods in Computer Vision*. IEEE, 2003.

[26] A. Tsai, T. Yezzi, and W. W. et.al, “A shape-based approach to the sementation of medical imagery using level sets,” *IEEE Trans. on Medical Imaging*, vol. 22, no. 2, pp. 137–153, 2003.