Estimation Theory and Tracking of Deformable Objects

Patricio A. Vela and Ibrahima J. Ndiour
School of Electrical and Computer Engineering
Georgia Institute of Technology
Atlanta, GA 30332-0250

***Abstract*— This paper provides a summary of several recently detailed recursive, filtering strategies as applied to the task of tracking deformable objects from a video sequence. Estimation theory provides the foundation for filtering objects based on presumed dynamics and (partial) uncertain measurements. All of the methods seek to arrive at an accurate contour signal in spite of measurement and prediction errors. They differ fundamentally in the state definition and in the estimation approaches.**

I. Introduction

The task of visually tracking objects in a video sequence is often decomposed into two parts: one for the rigid motion of the object and the second for the shape deformations of the object. The former, also called localization, is concerned with identifyng the gross position (and possibly scale and/or orientation) of the object for each incoming image in the sequence. The latter, often called segmentation, is concerned with identifying the bounding contour, or equivalently the binary silhouette, of the object for each image in the sequence.

Obtaining the bounding contour of an object, sometimes called *segmentation* is important for a variety of reasons. It is a fundamental component associated to many computer vision applications, which include missile tracking [33], cell tracking [8], full-body pose estimation [27], visual servoing [3], and time-sequenced medical imaging [32], amongst others.

Problems arise for algorithms requiring accurate boundary information when the boundary measurements are inaccurate. The inaccuracies arise for a variety of reasons, including simplistic or poor segmentation models, imaging noise, or conflicting image information. To overcome these problems, approaches have been developed to exploit the temporal nature of the visual signal being used for tracking. These approaches lie within the more general category of estimation theory as they are effectively *filtering* or *estimation methods* for visual tracking. For localization, which involves the estimation of the finite dimensional pose parameters, filtering can be achieved through a large variety of standard filtering approaches. For the segmentations, however, difficulties arise due to the infinite-dimensional manifold nature of the bounding contour. The difficulties have led to a variety of approaches for resolving the problem of uncertain shape measurements.

a) Early Approaches: Early approaches to contour-based segmentation, known as *snakes*, utilized spline-based parametrizations of closed curves. The polynomial coefficients of the splines formed a finite-dimensional state space for the evolving curve. Filtering methods were then applied such as Kalman filtering [22] or particle filtering as in the condensation filter [1].

There have been efforts to move beyond parametrized representations, such as the works in [2], [12], [19], [35]. This review paper will cover the methods found in [7], [9], [15], [16], [18], [25], [29], [32], all of which provide further reviews of the existing literature on estimation theory and visual tracking.

b) Volumetric Approaches: For sequences where the entire visual signal is available or for which a delay in estimation is allowed, volumetric approaches are applicable. These methods consider segmentation of the video stream to be a three-dimensional volumetric segmentation problem over the spatiotemporal volume. Many layered segmentation methods are volumetric in nature [30], [34]. Within the context of optimal filtering, [21] derive a method segmentation which requires access to the entire volume.

c) Tutorial Contents: This tutorial focuses on several recently developed recursive methods for curve estimation. While the evolving target is considered to be decomposed into both rigid and non-rigid components, only filtering of the non-rigid (or shape) substate will be emphasized. The rigid part, being a finite-dimensional substate, can be filtered using any finite-dimensional filtering strategy (Kalman, UKF, particle, etc.). Secondly, the technique used to generate the shape measurement. i.e., the segmentation algorithm, will not be discussed as there are many options for doing so [6]. The segmentation algorithm is a matter of design and should be determined according to the imagery. Section II covers filtering strategies based on finite-dimensional representations for the target shape space. Section III covers a local filtering method adapted to an implicit probabilistic curve representation. Next, Section IV covers methods that are derived by considering curve geometry and calculus. Finally, Section V details particle filtering methods that seek to move beyond finite-

dimensional curve parametrizations. The paper concludes with Section VI.

II. Low-Dimensional Parametrizations of Shape

For tracking problems whereby the shape deformations are known to lie in a low-dimensional subspace of shape space, dimension reduction tools from machine learning are applicable. Given that the possible shape variations of a tracked object live within a low-dimensional subspace, the subspace can be learned and modelled using principal component analysis (PCA) from a collection of sample shapes [31].

Consider an implicit representation for a closed-curve given the iso-contour of a scalar function $\phi : \mathbb{R}^2 \rightarrow \mathbb{R}$, for which a collection of representative sample shapes are available, $\mathcal{S} = \{ \phi_1, \cdots, \phi_{n_s} \}$ with n_s being the number of shapes. In practice, the implicit shape functions are defined on a discrete set of grid points, meaning that ϕ is approximated by a vector in $\mathbb{R}^N$ where N is the total number of grid points. Since N is large compared to the number of shapes provided, the application of PCA leads to a low-dimensional parametrization of the shape space,

$$\phi = \overline{\phi} + \sum_{i}^{n_e} \alpha_j \psi_j.$$

where $n_e \leq n_s$ is the number of shape variation eigenmodes retained, $\alpha_j \in \mathbb{R}$, $\overline{\phi}$ is the additive average of all of the implicit shape functions, and ψ_j are the principle components associated to the shape set $\mathcal{S}$. Rather than utilize the implicit representation ϕ for the low-dimensional shape set, an equivalent representation is the collection of coefficients that determine ϕ, $\alpha = (\alpha_1, ..., \alpha_{n_e})$. Collecting all of the eigenmodes ψ_i into a matrix $\mathbf{\Psi}$,

$$\phi = \overline{\phi} + \mathbf{\Psi}\alpha,$$

leads to the finite-dimensional parametrization, $\alpha \in \mathbb{R}^{n_e}$, of the shape space. Note that there is some freedom to choose the implicit function ϕ, with signed distance functions, binary functions, and density functions being feasible options [7], [9], [24].

The total state-space model for tracking includes both the rigid position and the shape. The dimension of the state is $n = n_r + n_e$ where n_r is the dimensions of the rigid sub-state and n_e is the dimension of the shape sub-state.

A. Unscented Kalman Filter

To perform temporal filtering of uncertain measurements under a finite-dimensional shape parametrization, [9] proposed the use of an unscented Kalman filter (**UKF**), which is a particular instantiation of a particle filter with Gaussian uncertainty.

Given an estimate of the mean state, $\widehat{\mathbf{x}}$, and the state covariance $\widehat{\mathbf{P}}$, the $2n+1$ points and weights to use in the particle filter are generated deterministically,

$$\begin{aligned}
\chi_o &= \widehat{\mathbf{x}}, \quad \omega_0 = \kappa/(n+\kappa), \\
\chi_i &= \widehat{\mathbf{x}} + \left(\sqrt{n+\kappa}\hat{\mathbf{P}}\right)_i, \ \omega_i = 1/(2(n+\kappa)), \\
\chi_{i+n} &= \widehat{\mathbf{x}} - \left(\sqrt{n+\kappa}\widehat{\mathbf{P}}\right)_i, \ \omega_i = 1/(2(n+\kappa)),
\end{aligned}$$

and are called *sigma points*. The recursive UKF algorithm begins with an estimate of the mean and covariance of the state and generates the sigma points.

Prediction: The next step is to propogate the particles forward under the state motion model, here denoted by f,

$$\chi_i{}^-(k) = f(\chi_i(k-1)), \quad i = 0, 1, ..., 2n+1,$$

where k denotes the current timstep and the $\cdot^-$ superscript denotes the estimate prior to correction. The predicted particles are used to reconstruct the predicted mean and covariance. For additive zero-mean Gaussian process noise with covariance $\mathbf{Q}$ independent of state, the predicted state mean and covariance are given by

$$\begin{aligned}
\widehat{\mathbf{x}}^-(k) &= \sum_{i=0}^{2n+1} \omega_i \chi_i{}^-(k) \\
\widehat{\mathbf{P}}^-(k) &= \sum_{i=0}^{2n+1} \omega_i \left(\chi_i{}^-(k) - \widehat{\mathbf{x}}^-(k)\right) \\
&\qquad \cdot \left(\chi_i{}^-(k) - \widehat{\mathbf{x}}^-(k)\right)^T + \mathbf{Q}.
\end{aligned}$$

Denote the measurement function by h. Let the measurement error be additive zero-mean Gaussian noise with covariance $\mathbf{R}$ independent of the process noise. The predicted mean and covariance of the measurement are

$$\begin{aligned}
\mathbf{y}(k) &= \sum_{i=0}^{2n+1} \omega_i h(\chi_i{}^-(k)) \\
\mathbf{P}_{\mathbf{yy}}(k) &= \sum_{i=0}^{2n+1} \omega_i \left(h(\chi_i{}^-(k)) - \mathbf{y}(k)\right) \\
&\qquad \cdot \left(h(\chi_i{}^-(k)) - \mathbf{y}(k)\right)^T + \mathbf{R}.
\end{aligned}$$

The predicted cross-correlation is

$$\mathbf{P}_{xy}(k) = \sum_{i=0}^{2n+1} \omega_i \left(\chi_i{}^-(k) - \widehat{\mathbf{x}}^-\right) \left(h(\chi_i^-) - \mathbf{y}\right)^T \quad (1)$$

Measurement Method: Classically measurements are directly obtained from sensors, however in the case of computer vision, the sensor measurement is an image from which the true position and shape measurements must be extracted. This may be written as $\mathbf{z} = h(I; \mathbf{x})$ which indicates that the measurement is generated from the image I, which itself depends on the actual state of the system $\mathbf{x}$. It is also common in computer vision for the measurement process to involve gradient-based optimization of an energy with local minima.

To avoid local minimia, [9] incorporated a multiple measurement strategy. Define the measurement function of

two arguments conditioned on the true state $h(\chi, I; \mathbf{x})$ to be the net result of applying the measurement optimization process to the image I using the state χ as an initial guess. Further, denote by $E(\chi, I; \mathbf{x})$ to be the final energy of the optimization process given by $h(\chi, I; \mathbf{x})$ (where lower is better). The multiple measurement strategy defines the actual measurement to be

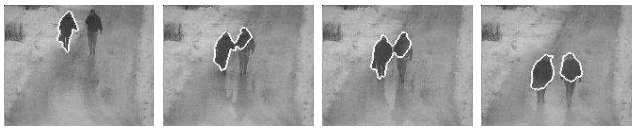
$$\mathbf{z} = \arg\min_{\chi \in \mathcal{M}} E(\chi, I; \mathbf{x}),$$

e.g., given mutiple intial guesses that are locally optimized, the measurement is chosen to be the one that is most optimal. In [9], a sigma-point cloud around the predicted rigid sub-state is chosen, $\mathcal{M} = \mathcal{M}(\widehat{\mathbf{x}}^-; \sigma)$, where $\mathcal{M}$ is a finite, discrete set of state elements. For a translation only rigid sub-state, $\mathcal{M}$ consists of 5 initial guesses.

Correction Step: The optimal gain for the current time step, given the predicted quantities, is $L = \mathbf{P}_{xy} \left(\mathbf{P}_{yy}\right)^{-1}$. Applying the gain to the predicted state and covariance given a mesurement z, leads to the updated estimates:

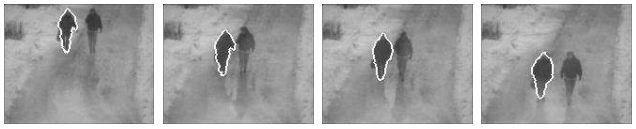
$$\widehat{\mathbf{x}}(k) = \widehat{\mathbf{x}}^-(k) + L\,(z - \mathbf{y})\,,$$
$$\widehat{\mathbf{P}}(k) = \widehat{\mathbf{P}}^-(k) - L\,\mathbf{P}_{yy}(k)\,L^T.$$

Example: Figures 1 and 2 depict the results of tracking a one person associated to a walking couple. The region-based method cannot track the left individual when the couple make contact, whereas the shape-based method is able to. Figure 3 shows that the method can handle partial occlusions of objects. Many active contour methods would fail to track across the occlusion or break the car into two bounding contours.



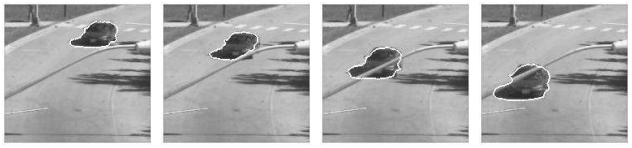
(a) Frame 3 (b) Frame 11 (c) Frame 13 (d) Frame 20

Fig. 1. Tracking a person with only region-based energy [9]



(a) Frame 3 (b) Frame 11 (c) Frame 13 (d) Frame 20

Fig. 2. UKF Tracking of person in shape-space [9].



(a) Frame 20 (b) Frame 34 (c) Frame 50 (d) Frame 65

Fig. 3. UKF Tracking of car in shape-space [9].

B. Bayesian Filtering in Shape Space

In the works [4], [5], [7], the shape space is parametrized using PCA and the shape dynamics of the system are learned by fitting the finite-dimensional shape evolution to a second-order autoregressive model,

$$\alpha(k) = \mu + A_1\,\alpha(k-1) + A_2\,\alpha(k-2) + \nu, \quad (2)$$

where ν is zero-mean Gaussian noise with covariance $\mathbf{Q}$. The model is learned using a few examplars of object motion. The probability of the current shape conditioned on the previous observed shapes is

$$Pr\left(\alpha(k) | \left\{\,\alpha(1), \ldots, \alpha(k-1)\,\right\}\right) \propto \exp\left(-\frac{1}{2} v^T \mathbf{Q}^{-1} v\right), \quad (3)$$

with $v = \alpha(k) - \mu - A_1\,\alpha(k-1) - A_2\,\alpha(k-2)$. Equations (2) and (3) consider a single motion model, whereas the work in [5] extends the problem to that of multiple models. If desired, the dynamics can incorporate the rigid body motion, otherwise generic or targetted motion models can be defined for the rigid sub-state.

Under some simplifying assumption (see [4]), estimation of the target state involves maximizing the conditional probability

$$Pr\left(\mathbf{x}(k) | I(k), \left\{\,\mathbf{x}(1), ..., \mathbf{x}(k-1)\,\right\}\right) \propto$$
$$Pr\left(I(k) | \mathbf{x}(k)\right) P\left(\mathbf{x}(k) | \left\{\,\mathbf{x}(1), ...\mathbf{x}(k-1)\,\right\}\right),$$

which for the second-order dynamical model, is equivalent to maximizing the probability

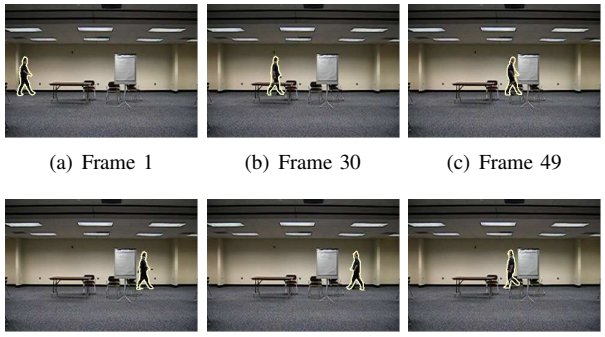
$$Pr\left(\mathbf{x}(k) | I(k), \mathbf{x}(k-2), \mathbf{x}(k-1)\right) \propto$$
$$Pr\left(I(k) | \mathbf{x}(k)\right)\; Pr\left(\mathbf{x}(k) | \left\{\,\mathbf{x}(k-2), \mathbf{x}(k-1)\,\right\}\right),$$

An equivalent optimization is obtained by seeking to minimize the negative log-likelihood of the probabilities, where the negative log-likelihoods are now represented by energies

$$E\left(\mathbf{x}(k), I(k); \mathbf{x}(k-2), \mathbf{x}(k-1)\right) =$$
$$E_{meas}(\mathbf{x}(k); I(k)) + \nu E_{shape}(\mathbf{x}(k); \mathbf{x}(k-1), \mathbf{x}(k-2)),$$

where ν is a scalar variable for adjusting the relative weighting between the two energies. The scalar ν behaves somewhat similarly to a gain variable that modulates between trust in the measurement versus trust in the predicted shape. The data matching energy can be any sensible energy suited to the image sequence. The shape energy follows naturally from Eq. (3).

The references [4], [5] both utilized a gradient descent procedure to perform energy minimization with a signed-distance function as the implicit shape descriptor. More recent work [7] utilizes an implicit probabilistic representation, which models the shape as an iso-contour of a two-dimensional scalar function whose domain is $[0, 1]$. The convexity of the representation under addition is exploited to derive a global optimization strategy, relying on iterated



(a) Frame 1 (b) Frame 30 (c) Frame 49

(d) Frame 70 (e) Frame 137 (f) Frame 164

Fig. 4. Shape-based tracking with global optimization [7].

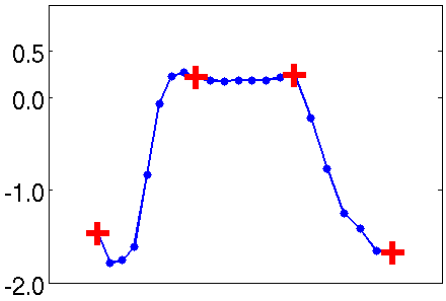


Fig. 5. Using optimal matching energy to identify occlusions [7].

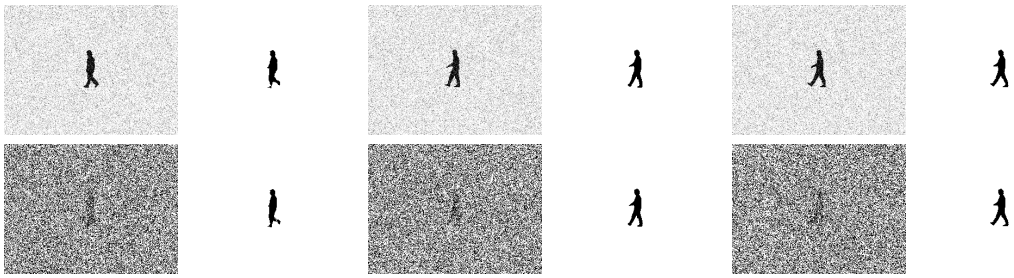


Fig. 6. Optimal estimates for two different Gaussian noise levels ($\sigma \in 64, 512$) [26].

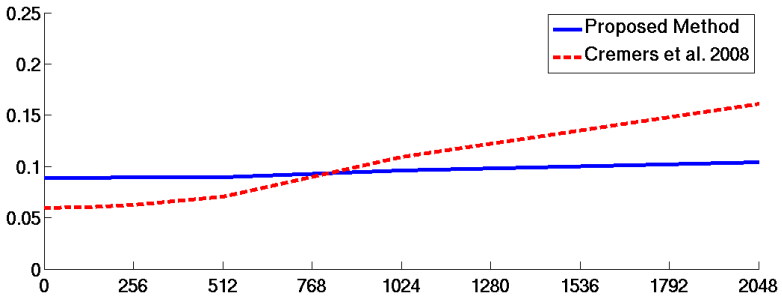


Fig. 7. Average segmentation error as a function of noise level ([4] in red dashed line vs. [26] in blue solid line).

convex projections, for identifying the location and shape of the target in an image frame. Results of the procedure are shown in Figure 4, where a target is partially occluded by a table and fully occluded by presentation board. Due to the global optimization, the tracking algorithm does not lose the target. Further, the optimal energy value of the gloal optimization can be used to identify when measurements are invalid. Figure 5 depicts the globally optimal energy of the track sequence from Figure 4 for the time period around the first full occlusion. The jump in energy is indicative of a complete occlusion.

In [26], the finite-dimensional nature of the shape space is exploited to identify a closed-form optimal filtering solution, overcoming some of the key limiting assumptions from [4]. Due to the fact that the closed-form solution optimizes over a longer trace of the shape history, the method is more robust to noise. Robustness to noise versus the original approach can be seen in Figure 7, where the closed-form solution maintains an error that is independent of the noise level.

III. Implicit Probabilistic Representation

This section revisits the implicit represention of closed curves by the iso-contour of a density function whose values lie in the range $[0, 1]$. Such a representation was considered in [7] within in a shape-constrained setting (as briefed in §II-B). It is also considered in [15], [18], [33] within an unconstrained setting. This section summarizes the unconstrained setting, covering both a general purpose strategy with fixed gains and a specific strategy with optimized gains given Gaussian uncertainty.

A. Geometric Filtering

The geometric filtering method to be described here presume the existence of a measurement strategy whose seg-

TABLE I
STATE MOTION MODELS

Static prior	$\begin{cases} \dot{g} = 0 \\ \dot{P} = 0 \end{cases}$
Constant group velocity	$\begin{cases} \dot{g} = \xi, \quad \dot{\xi} = 0 \\ \dot{P} = 0 \end{cases}$
Constant velocity (1)	$\begin{cases} \dot{g} = \xi, & \dot{\xi} = 0 \\ \dot{P} + \nabla P \cdot X = 0, & \dot{X} = 0 \end{cases}$
Constant velocity (2)	$\begin{cases} \dot{g} = \xi, & \dot{\xi} = 0 \\ \dot{P} + \nabla P \cdot X = 0, & \dot{X} + \nabla X \cdot X = 0 \end{cases}$

mentation is determined by two competing probabilities, Pr_F and Pr_B, denoting the foreground and background, respectively. Prior to segmentation, the two probabilities are typically normalized so that $Pr_B = 1 - Pr_F$.

Assume that the image data is corrupted with noise and that the measurement models for the foreground and background are approximate (hence introduce additional uncertainty). Also, assume that the uncertainty affects the computation of the foreground/background probabilities in a multiplicative fashion,

$$Pr_F = Pr_F^* \, \eta, \tag{4}$$

where Pr_F^* denotes the true probability, and a similar equation holds for Pr_B. Transformation by the logarithm converts the multiplicative error into additive error,

$$\log(Pr_F) = \log(Pr_F^*) + \log \eta.$$

whereby standard linear filtering can be applied, pointwise. In the original space, the additive filtering results in geometric filtering. This section will propose a filtering strategy for the track state, which will involve pointwise geometric filtering of probabilistic shape sub-state. A complete model of the state includes both position and velocity information, $\mathbf{x} = (g, P, \xi, X)$, where (g, ξ) is the rigid sub-state and (P, X) is the shape sub-state.

Prediction: The prediction step is best chosen to represent the actual physical dynamics of the target. Precise knowledge of the target dynamics may not be available, in which case Table I provides a collection of general purpose static and dynamic motion models which should provide a decent approximation.

Model Measurement: The model measurement is obtained by extracting the components of the internal state model that are equivalent to those obtained from the sensor measurement. Typically this will be $(\widehat{g}^-, \widehat{P}^-)$ but can also include the shape velocity $\widehat{X}^-$.

Sensor Measurement: Measurement of the state minimally involves estimation of the rigid pose parameters and also the shape of the target, hence a localization and segmentation strategy will be needed. Once localization and segmentation are performed on the current image, a registration procedure is applied to match the resulting measured probability field with the predicted probability field, yielding a measurement g_m for the group motion and the measurement P_m for the shape. If desired, the velocity field X_m can be measured by computing the optical flow [11] between two subsequent aligned images. In practice, the group velocity ξ is not directly measurable. The measurement proper can utilize any segmentation technique so long as the final result is of the form of an implicit probability field.

Correction: Correction on the probability field is achieved through geometric averaging. Given the predicted probability field, $\widehat{P}^-$, and the current measured probability field, $\widehat{P}_m$, the current corrected probability field, $\widehat{P}$, is

$$\widehat{P}(r) = \left(\widehat{P}^-(r)\right)^{1-K_{xx}} \left(\widehat{P}_m(r)\right)^{K_{xx}},$$

where K_{xx} varies in the range $[0, 1]$ and is chosen based on uncertainty estimates of the measurement and the prediction. Low K_{xx} is biased towards the predicted probability, high K_{xx} is biased towards the measured probabilities. This method works when the prediction and measured probability fields dot not differ radically.

When the two fields have sufficient disparity, the geometric averaging technique no longer works due to non-local shape effects. Instead, an error vector field X_{err} needs to be computed between the predicted and measured densities. Flowing along the error vector field should take P^- to P_m in unit time. The error vector field between the two densities can be computed using a variety of methods, such as optical flow, displacement flow or optimal mass transport [20], [23]. The correction is then given by

$$\widehat{P}^+ = \Phi_{K_{xx}}^{X_{err}}(\widehat{P}^-),$$

where $\Phi_{K_{xx}}^{X_{err}}$ denotes the flow along X_{err} for time K_{xx}. In practice, we did not find this to be necessary.

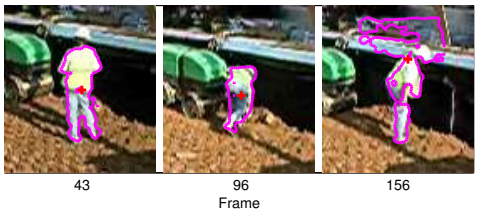
The velocity field is much simpler to correct on since it is a vector space. There are two ways to induce a correction on the velocity field, one is through an error in the measured probabilities and the other is through an error in the shape velocities:

$$X^+ = X^- + K_{vx} X_{err}(P_m, P^-) + K_{vv}(X_m - X^-)$$

where $X_{err}(P_m, P^-)$ is as defined above. The parameters K_{vx} and K_{vv} vary in the range $[0, 1]$ and are chosen according to uncertainty estimates of the measurement and prediction.

Example: Figure 8 depicts the tracking results for an image sequence which contains measurement error due to overlapping distributions. The frames, from left to right, depict the lowest error segmentation, a typical segmentation, and the largest error segmentation (misclassified pixels). The overlapping distributions cause the measurement error to be large on a few frames. The measurement error arises from regions of the image that have intensities at the decision boundary, which causes the segmentation

(a) Snapshot



(b) Deformotion

(c) Probabilistic Contour Observer

Fig. 8. Tracking results for sequence with overlapping distributions [18].

boundary to fluctuate by a large amount. Notice that the deformotion filter [12], which is strictly contour-based has difficulties recovering. The probabilistic method does not blow-up. Since the probabilistic estimator maintains a history of all interior and exterior pixels, uncertainty in the boundary region is temporally smoothed.

B. Optimal Geometric Filtering

This section consider a more explicit measurement model, Bayesian segmentation, and derives an optimal filtering strategy given Gaussian uncertainty. Additional details can be found in [18].

Consider an image I defined over a compact domain of the plane and taking values in $\mathbb{R}$. Further, assume that measurement of the pixel intensities has been corrupted by additive Gaussian noise ν with zero mean and variance σ_ν^2. Classification is performed through Bayesian segmentation [10] with two classes: foreground and background. The two classes are modelled with a Gaussian distribution for the pixel intensities. Assuming uniform priors and a normal distribution $\mathcal{N}(\mu_F, \sigma_F^2)$ for the foreground pixels, then the measured likelihood for the corrupted pixel $I(r)$ to be classified as foreground is given by:

$$\zeta_F(r) = \sqrt{\delta} \cdot e^{-\frac{1}{2}\left(\frac{I(r)+\nu(r)-\mu_F}{\sigma_F}\right)^2},$$

where δ is a positive normalizing factor. The expression for the measured likelihood can be expanded further:

$$\zeta_F(r) = \sqrt{\delta} \cdot e^{-\frac{1}{2}\left(\frac{I(r)-\mu_F}{\sigma_F}\right)^2} e^{-\frac{1}{2}\left(\frac{\nu(r)}{\sigma_F}\right)^2} e^{-\left(\frac{\nu(r)\,(I(r)-\mu_F)}{\sigma_F^2}\right)}, \tag{5}$$

which can be rewritten as

$$\zeta_F(r) = P_F(r) \cdot \eta(r; \mu_F, \sigma_F),$$

where $P_F(r)$ consisting of the first two terms from (5) is the true classification likelihood, and $\eta(r; \mu_F, \sigma_F)$ consisting of the remaining two terms is the class measurement noise. A similar derivation holds for the background classification densities. Corruption by additive noise on the image data results in multiplicative uncertainty for the foreground/background likelihoods.

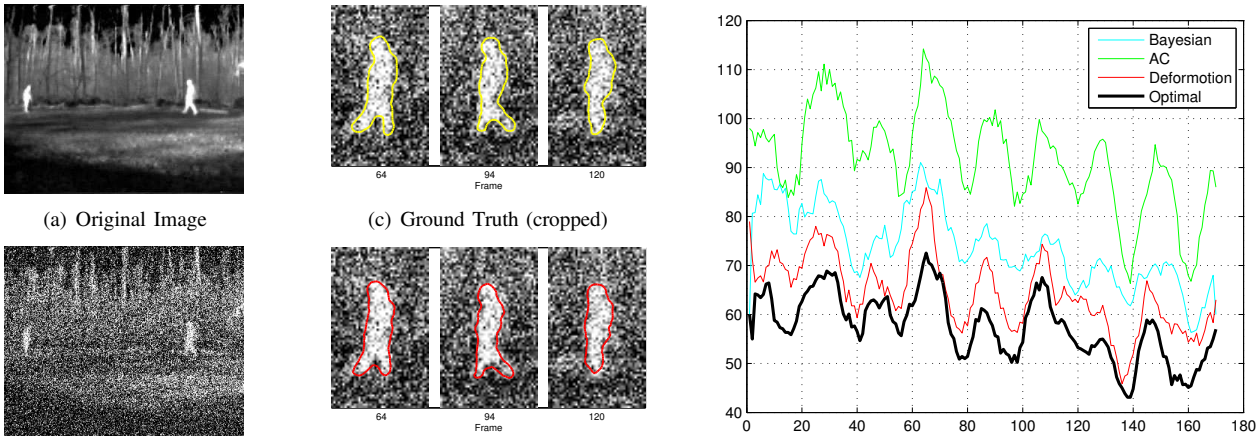
TABLE II
FILTERING EQUATIONS FOR VISUAL TRACKING

Prediction	$\begin{cases} \hat{\rho}_k^- = \hat{\rho}_{k-1}^+ \\ \hat{P}_k^- = \hat{P}_{k-1}^+ + Q \end{cases}$
Update	$\begin{cases} K_k = \hat{P}_k^- (\hat{P}_k^- + R)^{-1} \\ \hat{\rho}_k^+ = \left(\hat{\rho}_k^-\right)^{1-K_k} \cdot (\zeta_k)^{K_k} \\ \hat{P}_k^+ = (1-K_k)^2\, \hat{P}_k^- + K_k^2\, R \end{cases}$

Filtering: As noted in Section III-A, geometric filtering is the net result of seeking a mechanism to temporally filter the uncertain segmentations. When coupled with the Bayesian segmentation algorithm, an explicit method of the optimal pointwise geometric filter is derivable [18]. Endowing each pixel probability ρ with a covariance P, leads to the prediction and update equations in Table II, where the subscript k indicates the discrete time step.

Based on the description of the design, the optimal estimation algorithm can be summarized as follows:

1) Estimate the additive imaging noise prior to the visual tracking process.
2) For every pixel, run two filters to estimate the foreground and background likelihoods ($\hat{P}_F(r)$ and $\hat{P}_B(r)$):
   a) at the prediction step, run the corresponding equations in Table II to obtain the predictions.
   b) obtain a measurement by taking the classification likelihood given by Bayesian segmentation



(a) Original Image (c) Ground Truth (cropped)

(b) Corrupted Image (d) Optimal Estimator (cropped) (e) Comparison (# of Misclassified Pixels vs. Time)

Fig. 9. Optimal point-wise target estimation using geometric filtering of probabilistic contours [18].

on the current image.

c) at the update step, run the corresponding equations in Table II to obtain the updates.

3) The estimated classification probability field is obtained by normalizing the likelihood estimates previously obtained: $\frac{\hat{P}_F}{\hat{P}_F+\hat{P}_B}$. The 50% contour of this probability field defines the bounding contour of the target.

Example: Figure 9 depicts a tracking scenario corrupted with Gaussian noise. A comparison of the misclassified pixels per frame shows that the optimal filter for the implicit probabilistic shape representation is able to optimally filter out the temporally uncertainty. The comparison methods, active contour (AC) and deformotion, were manually tuned for optimal performance.

IV. Curve-Space Representations

For object shapes that are not constrained to lie in a lower-dimensional submanifold, approaches that are consistent with the manifold structure have been developed. These are methods that attempt to derive as much of the underlying mathematics within the space of curves, then seek the appropriate representation and algorithms that can reproduce the derived mathematics. The infinite-dimensional manifold structure of the space of smooth, closed planar curves means that there is no unique structure to the shape description, and consequently not a unique approach to filtering.

A. Local Kalman Filtering Using Transverse Coordinates

The curve estimation method described in this section relies on the existence of a local error metric between two smooth, closed planar curves $\mathcal{C}_0$ and $\mathcal{C}_1$. Given the two curves, a characteristic error vector field X_{char} can be obtained by solving a series of Laplace and Poisson equations with boundary conditions [20]. The characteristic curves of the error vector field are non-intersecting, they approach the curves $\mathcal{C}_0$ and $\mathcal{C}_1$ from normal directions, and their speed never vanishes. For a particle $\boldsymbol{x}_0 \in \mathcal{C}$, its traveling distance, d, at position $\boldsymbol{x}$ along the characteristic of X_{char} through $\boldsymbol{x_0}$ is defined as the arc-length of the characteristic curve connecting $\boldsymbol{x}_0$ and $\boldsymbol{x}$. Measuring these traveling distances from a complete set of initial locations is obtained by solving

$$\begin{cases} d(\cdot,0) = 0, \\ d_\tau + X_{char}^T \nabla_x d = 1, \end{cases}$$

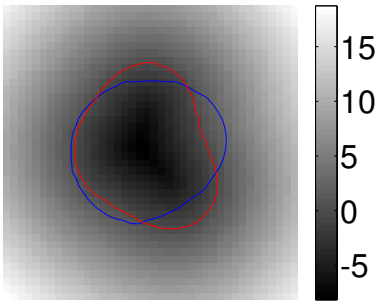
where τ is an artificial time parameter for the PDE equation and $d : \mathbb{R}^2 \times \mathbb{R}^+ \to \mathbb{R}$. Figure 10(a) an example of the distance map obtained, whiile Figure 10(b) depicts the transverse curves for segments of two curves.

For the curve $\mathcal{C}$ local to $\mathcal{C}_0$ and $\mathcal{C}_1$, the traveling distance map and its characteristics define a 1D family of transverse coordinate frames. Let s be the arc-length parameter of the curves in question, and assume that $\mathcal{C}_0(0)$, $\mathcal{C}_1(0)$ and $\mathcal{C}(0)$ lie on the same characteristic. Then at $\mathcal{C}(s)$ the associated transverse curve intersects, for example, the curves $\mathcal{C}_0$ and $\mathcal{C}_1$, at the same arc-length parameter s. The coordinate location on the transverse curve of the curve point $\mathcal{C}_0(s)$ is given by

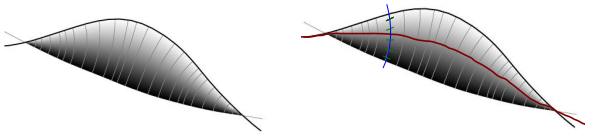
$$sd(\mathcal{C}_0(s);\mathcal{C}(s)) = d(\mathcal{C}_0(s),0), \quad \text{with } \mathcal{C} = d^{-1}(\cdot,0),$$

where $sd(p;\mathcal{C})$ describes the signed traveling distance from a point $\boldsymbol{x}$ to its corresponding point on $\mathcal{C}$. The distance is negative when $\boldsymbol{x}$ lies interior to $\mathcal{C}$ and positive exterior to $\mathcal{C}$. Figure 10-(b) depict the coordinate system (in blue) along one of the transverse curves (where $\mathcal{C}$ is in red).

The goal of the filtering process is to arrive at estimates of the curve and its normal velocity, denoted by $\hat{\mathcal{C}}$ and $\hat{\beta}$, respectively. To explicitly exploit the fact that these transverse curves have their own coordinates, we will specify the coordinates on the transverse line of a particular curve point $\hat{\mathcal{C}}(s)$ by $x(s)$, whose value is given by $x(s) = sd(\hat{\mathcal{C}}(s);\mathcal{C}(s))$, where $\mathcal{C}(s)$ is presume to be the true curve

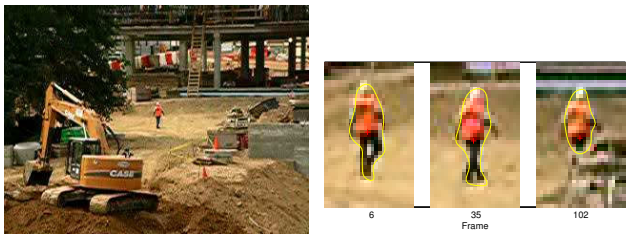


(a) Distance map for red curve given a comparison curve.



(b) Transverse curves. (c) Transverse coordinates.

Fig. 10. The topology and geometry of curve comparison and transverse curve coordinates. The red curve in (c) is the true curve.



(a) Sample image. (b) Ground Truth

(c) Active Contour (d) Local Kalman filter

Fig. 11. Tracking snapshots and comparison for baseline active contour measurements versus local Kalman filtered estimates [16].

state. This notation is called the point-notation for the curve. In addition to the curve states, the rigid motion of the curve also forms part of the state, $\mathbf{x} = (g, \mathcal{C})$. If the rigid velocity is included, then $\mathbf{x} = (g, \mathcal{C}, \xi)$. for the Kalman filter, the state covariance will also be modelled and propogated. To explicitly indicate that the curve is time varying consider the functional dependence to be $\mathcal{C}(k, s)$ where k is the time-step and s is the arc-length parametrization. For simplicity, the arc-length parametrization will sometimes be omitted.

Prediction: The work in [16] applies only to a static prediction model on the curve, which means that:

$$\widehat{\mathcal{C}}^-(k) = \widehat{\mathcal{C}}(k-1),$$
$$\widehat{P}^-(k) = \widehat{P}(k-1) + Q,$$

where $\widehat{P}$ is the estimated curve covariance (defined along the curve). Work in [17] extends the state space to the case of dynamically evolving curves. The rigid sub-state $\widehat{g}$ or $(\widehat{g}, \widehat{\xi})$ evolves as first or second order system, respectively.

Model Measurement: The predicted measurements are extracted directly from the predicted state. Due to the available sensor measurements, the measurement model extracts at most the rigid state and the curve, $(\widehat{g}, \widehat{\mathcal{C}})$.

Sensor Measurement: The sensor measurement follows similarly to the probabilistic contour approach (§III), where measurement of the state minimally involves estimation of the rigid pose parameters and also the shape of the target through a localization and segmentation strategy. Registration is needed to match the resulting contour to the predicted contour yielding a measurement g^m for the group and the measurement $\mathcal{C}^m$ for the shape.

Correction: As with the probabilistic contour, the group correction is a finite-dimensional state and can therefore be filtered using standard methods with relative ease. Depending on the prediction motion model chosen (linear vs. nonlinear), finite-dimensional filtering such as Kalman, extended or unscented Kalman are appropriate to use in order to correct the measured group.

Correction on the shape requires the construction of the transverse coordinates, which requires the characteristic vector field using the predicted and measured curves. The correction applied to the curve, in point notation, is then

$$\hat{x}(s) = \hat{x}^-(s) + K(x^m(s) - \hat{x}^-) = Kx^m(s),$$

since the predicted curve is at the origin. The curve state $\hat{\mathcal{C}}(k, s)$ is uniquely reconstructed by $\hat{x}(s)$. The gain chosen is the optimal gain

$$K = \widehat{P}^-(k)\left(\widehat{P}^-(k) + R\right)^{-1},$$

where R gives the curve measurement covariance. Associated to this gain is the covariance update

$$\widehat{P}(k) = \widehat{P}^-(k)\,(1 - K)$$

Example: Figure 11 depicts a sample frame and the tracking results for the base measurement strategy utilizing active contours plus the local Kalman filter. The ground truth segmentations are also provided for comparison. Table III gives quantitative tracking results for a variety of tracking algorithms, each comparison algorithm is manually optimized. Strikeout indicates loss of track, in which case the quantitative results are until shortly prior to track loss. Of note, only the local Kalman filter is able to track the entirety of the sequence.

B. Filtering Using Curve Geodesics

The work in [28] examines the underlying geometry of the space of smooth, closed planar curves (modulo

Metric\Algorithm		Active Contour	Deformotion	Shape	Observer
Trackpoint error	(L_2/L_∞)	~~2.2/6.6~~	~~2.2/9.6~~	~~7.6/18.6~~	1.8/6.2
# Misclassified Pixels	(avg/max)	~~78/202~~	~~72172~~	~~87/160~~	63/111
Mean Laplace	(avg/max)	~~1.0/3.7~~	~~0.9/3.1~~	~~3.4/8.4~~	0.7/1.3
Max. Laplace	(avg/max)	~~2.9/8.9~~	~~2.3/7.9~~	~~3.4/8.4~~	2.0/3.5
# Frames Tracked		~~109~~	~~109~~	~~115~~	350

TABLE III

PERFORMANCE TABLE USING QUANTITATIVE COMPARISON METRICS [16].

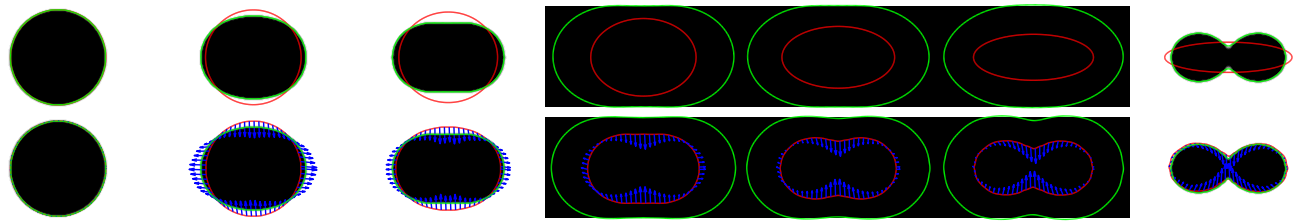


Fig. 12. Tracking a synthetic deforming circle through a total occlusion. This experiment demonstrates the need for the dynamical model that extrapolates shape. In the first few frames, where there is no occlusion, the image segmentation (green) alone correctly follows the shape evolution. However, when the occlusion appears, the image segmentation is uninformative, and the dynamical model extrapolates the shape (red) and velocity (blue) of the contour (middle frames). A dynamical model with only affine motion (top row) cannot extrapolate the deformation. The infinite-dimensional model, on the other hand, correctly predicts the evolution towards a bi-lobate shape. Red: $\widehat{\mathcal{C}}^-(k)$; blue: $\widehat{\beta}^-(k)$ and green: $\mathcal{C}^m(k)$ [28].

reparametrizations) to arrive at a manifold and geodesic structure for this space. The essential contribution revolves around the calculus of a Sobolev-type, Riemmanian metric for the space of smooth, closed curves, and builds on recent developments regarding the geometry of closed curves [13], [14]. Through the Sobolev-type metric, geodesic paths connecting curves are found. The paths are defined modulo curve translation and scale changes, which leads to a representation for the curves as elements of the Stiefel manifold $\mathrm{St}(2, V_{od})$, where $V_{od} \equiv \{ f : \mathbb{S}^1 \to \mathbb{R} \mid f(0) = -f(2\pi) \}$.

The curve filter designed using curve geodesics considers a constant velocity prediction model with perturbation noise at the velocity level, and a curve measurement model with noise. The curve estimate is given by $\widehat{C} \in B$ and the tangent to the curve is given by $\widehat{\beta} \in T_{\widehat{C}}B$. The manifold B is the space of smooth, closed planar curves modulo reparametrizations. Corrections will occur at the velocity level, although [28] do briefly describe how to perform corrections at the curve level. The procedure below summarizes the resulting Luenberger observer on B; for implementation details, see [28].

Prediction: Exploiting the geodesic structure defined on B, the dynamical prediction model is

$$\widehat{\mathcal{C}}^-(k) = \exp_{\widehat{\mathcal{C}}(k-1)} \left(\widehat{\beta}(k-1) \right),$$
$$\widehat{\beta}^-(k) = P_{\widehat{\mathcal{C}}(k-1), \widehat{\mathcal{C}}^-(k)} \left(\widehat{\beta}(k-1) \right),$$

where the exp is defined by geodesic flow of the curve along the curve tanget $\beta(k-1)$, and $P_{\widehat{\mathcal{C}}(k-1), \widehat{\mathcal{C}}^-(k)}$ indicates parallel transport of the tangent vector along the geodesic connecting the curve $\widehat{\mathcal{C}}(k-1)$ to $\widehat{\mathcal{C}}^-(k)$.

Model Measurement: The model measurement is the curve proper $\widehat{\mathcal{C}}^-(k)$.

Sensor Measurement: The measured curve is obtained through any segmentation algorithm capable of providing a closed curve indicating the boundary of the tracked object. The measurement is denoted by $\mathcal{C}^m(k)$.

Correction: The correction for the curve is applied at the curve velocity level, which will induce a correction on the state during the prediction step. It is given by

$$\widehat{\mathcal{C}}(k) = \widehat{\mathcal{C}}^-(k),$$
$$\widehat{\beta}(k) = \widehat{\beta}^-(k) + K \log(\widehat{\mathcal{C}}^-(k), \mathcal{C}^m(k)),$$

where the log term gives the tangent vector element whose geodesic flow goes from $\widehat{\mathcal{C}}^-(k)$ to $\mathcal{C}^m(k)$, and $K \in \mathbb{R}$ is the correction gain.

Examples: To visualize the benefits of evolving the curve, rather than considering a static curve undergoing a group transformation, consider the two scenarios depicted in Figure 12. During the period of the occlusion, there are no measurements to correct the predictive models. The affine transformation model cannot adequately predict the changes in the curves, whereas the prediction model based on geodesic flow can more accurately predict the curve state after the occlusion ends. Figure 13 considers the underwater visual tracking of a flatworm.

V. PARTICLE FILTERING

Particle filtering on the shape manifold poses a few difficulties, the primary of which is the dimensionality of the state space. This section presents two strategies,

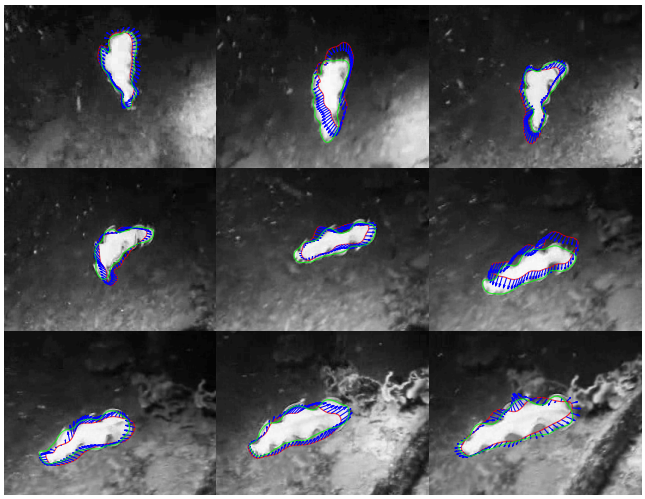


Fig. 13. Tracking a flatworm (left to right, top to bottom) using the proposed filtering technique: the red curve is $\widehat{\mathcal{C}}^-(k)$ the blue arrows are $\widehat{\beta}^-(k-1)$, and the green curve is the measurement $\mathcal{C}^m(k)$. This experiment demonstrates the dynamics of the contour and deformation under the constant velocity plus perturbation model, which correctly models the dynamics of the flatworm [28].

one which implements a complete particle filter for the rigid pose sub-state and a mode-based particle filter for the shape sub-state, and the second which approximates the local deformations with a finite-dimensional basis that is updated over time. The two techniques implement a full particle filter on a sub-state and a posterior mode particle filter on the complementary sub-state. The posterior mode filter replaces all particle estimates of the sub-state with the current mode as the correction step. Details can be found in [25], [32].

First, a state model will be described, followed by the specific instantiations associated to the two particle filters. The general state model consists of a closed planar curve $\mathcal{C} : \mathbb{R} \to \mathbb{R}^2$, a group transformation g, and also a shape deformation along the curve normal $\beta : \mathbb{R} \to \mathbb{R}$ defined on the curve. Note that the group transformation acts on the curve as $g \cdot \mathcal{C}$, which effectively transforms the curve points according to the group motion (translation, $SE(2)$, affine, etc.). As defined, the deformations will naturally transform appropriately since the curve deformation is constructed using $v = \beta\mathcal{N}$, where $\mathcal{N}$ is the curve normal.

A. Affine, Posterior Mode Particle Filtering

In [25], the group transformation is affine transformation and the shape deformations are neglected, thus $\mathbf{x} = (g, \mathcal{C})$. A particle filter is defined on the group sub-state (affine transformations), while a posterior mode tracker is defined on the shape sub-state (level-set representation). Beginning with a collection of samples that represent the current state distribution, the affine, posterior mode particle filter proceeds as follows:

Prediction: All of the particles need to be propogated under the motion model, which in this case involves dynamic propogation of the group motion with a static shape prior,

$$\widehat{g_i}^-(k) = f(\widehat{g}_i(k-1)), \quad \widehat{\mathcal{C}_i}^-(k) = \widehat{\mathcal{C}}_i(k-1),$$

followed by the affine transformation of the curve

$$\widehat{\mathcal{C}_i}^-(k) = g_i^-(k) \cdot \widehat{\mathcal{C}_i}^-(k).$$

Model Measurement: The predicted measurement density is the collection of estimated curves $\widehat{\mathcal{C}_i}^-$ and their weights, ω_i.

Sensor Measurement: The collection of predicted particles is used to generate the sensor measurement given an image. In [25], the proposed sensor measurement involves running an iterative optimization-based segmentation strategy for a limited amount of iterations (not to convergence). The function to minimize is the energy functional $E_{meas}(\mathcal{C}; I)$, which defines an error energy associated to the curve $\mathcal{C}$ given the sensed image I. The segmentation strategy is run for each particle state using the particle contour $\widehat{\mathcal{C}}_i$ as the initial condition for the iterative algorithm, leading to the measurement $\mathcal{C}_i^m$.

Correction: To perform correction, the weights are recomputed using importance sampling as per,

$$\tilde{\omega}_i(k) = \exp\left(-E_{meas}(\widehat{\mathcal{C}}_i(k), I(k))/\sigma_d^2\right) \cdot \exp\left(-d^2(\mathcal{C}_i(k), \widehat{\mathcal{C}}(k-1))/\sigma_p^2\right),$$

where $d(\widehat{\mathcal{C}}_i(k), \mathcal{C}(k-1))$ is a discrepancy measure on the space of curves (set symmetric difference, Haussdorf distance, etc.) from which the updated weights are

$$\omega_i(k) = \tilde{\omega}_i(k) / \sum_{j=1}^{N} \tilde{\omega}_j(k).$$

The sample with the largest weight is the estimated curve for the current timestep. Prior to repeating the procedure for the next timestep, the distribution should be resampled, according to the current estimated distribution, to obtain N particles each with the same weight $1/N$.

Examples: Figures 14 and 15 depict two tracking scenarios. In both cases there is significant interframe motion, while the van scenario also includes occlusion of the track target. Comparison with the condensation filter in [25] shows that the affine, posterior mode filter is quite competitive, requiring fewer particles to successfully track.

B. Particle Filtering on Highly Deforming Shape Spaces

The posterior mode tracker described in §V-A has been extended in [32] through the consideration of finite-dimensional parametrized models for the shape deformations. In particular, the work presumes that β can be approximated through a finite collection of control points on the curve for which the normal deformation is known; the deformations over the entire curve are obtained through B-spline interpolation. Furthermore, the control points are

Fig. 14. Van Sequence: Tracking of target with occlusion and scale variation [25]

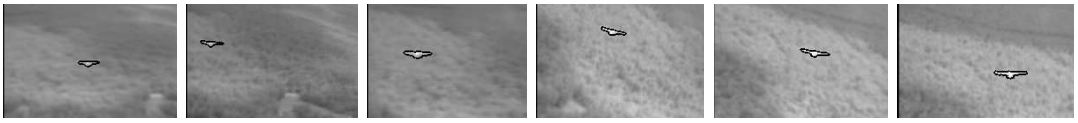


Fig. 15. Plane Sequence: Tracking with 30 particles. Images have been cropped for better visualization [25].

broken up into two disjoint sets forming complementary vector spaces, $\beta = \beta_s \oplus \beta_r$, one of which models the dominant shape variation and the other which models small, normally distributed shape variations.

The state space of the system is augmented appropriately to be $\mathbf{x} = (g, \mathcal{C}, \beta_s, \beta_r)$, where β_s are the parameters of the effective basis deformation, while β_r are the parameters of the residual basis deformation. Together, β_s and β_r, define the vector β which is used to reconstruct the shape deformation of the curve $\mathcal{C}$ (behaving, in a sense, much like a finite dimensional approximation of the curve velocity state). While [32] spends some time discussing parametrization choices for the deformations, this review paper will simply state that any choice of finite-dimensional interpolation strategy will work and focus instead on the filtering strategy required.

The state space is partitioned into two components, one of which is presumed to be affected by zero-mean Gaussian noise only, and the other which evolves in time and is perturbed by zero-mean Gaussian noise. The former is $(\mathcal{C}, \beta_r)$ and the latter is (g, β_s). For the former sub-space, a posterior mode tracker will be applied, whereas the latter will be estimated using a full particle filter. Consider a collection of estimate states $\mathbf{x}_i$ with associated weights ω_i used to generate the state distrubution. The estimate is propogated in time as follows:

Prediction: The prediction equations for the particles are:

$$\begin{aligned}
\widehat{g}_i^-(k) &= A_g \widehat{g}_i(k-1), \\
\widehat{\beta}_{s,i}^-(k) &= A_{\beta,s} \widehat{\beta}_{s,i}(k-1), \\
\beta_{r,i}(k) &= \beta_{r,i}(k-1), \\
\widetilde{\mathcal{C}}_i(k) &= \widehat{\mathcal{C}}_i(k-1) + \widehat{g}_i(k) + \widehat{\beta}_s \mathcal{N}_s, \\
\widehat{\mathcal{C}}_i^-(k) &= \widetilde{\mathcal{C}}_i(k) + \beta_{r,i} \mathcal{N}_r,
\end{aligned}$$

where $A_\cdot$ represents a linear transition matrix for updating the finite-dimensional sub-states, and $\mathcal{N}_\cdot$ is the normal vector to the curve associated to the appropriate shape deformation basis.

Model Measurement: The predicted measurement density is the collection of estimated curves $\widehat{\mathcal{C}}_i^-$ and their weights ω_i.

Sensor Measurement: The collection of predicted particles is used to generate the sensor measurement given an image. Measurements are obtained by running an iterative optimization-based segmentation strategy for a limited amount of iterations (not to convergence). In [32] the energy functional $E_{meas}(\mathcal{C}; I)$ to minimize includes both region-based terms and edge-based terms. The segmentation strategy is run for each particle state using the particle contour $\widehat{\mathcal{C}}_i$ as the initial condition for the iterative algorithm, leading to the measurement $\mathcal{C}_i^m$.

Correction: To perform correction, the weights are recomputed using importance sampling as per,

$$\begin{aligned}
\tilde{\omega}_i(k) = \omega_i(k-1) \cdot \exp\left(-E_{data}(\widehat{\mathcal{C}}_i(k); I(k))/\sigma_d^2\right) \\
\cdot \exp\left(-d^2(\mathcal{C}_i(k), \widehat{\mathcal{C}}(k-1))/\sigma_p^2\right),
\end{aligned}$$

where $d(\widehat{\mathcal{C}}_i(k), \mathcal{C}(k-1))$ is a discrepancy measure on the space of curves (set symmetric difference, Haussdorf distance, etc.) from which the updated weights are

$$\omega_i(k) = \tilde{\omega}_i(k) / \sum_{j=1}^{N} \tilde{\omega}_j(k).$$

The sample with the largest weight determines the corrected estimate for the $(\mathcal{C}, \beta_r)$ sub-state. All particle estimates for this sub-state are replaced by the estimated mode. Prior to repeating the procedure for the next timestep, the distribution should be resampled, according to the current estimated distribution. For the new set of N particles, each particle has the same weight $1/N$.

Example: Figure 16 depicts a tracking scenario for the posterior mode particle filter with parametrized deformations. It does not get confused by the background circle in spite of the false edge information and umodelled color distribution.

VI. Conclusion

This paper reviewed recent approaches to estimating the evolving shape of a tracked deformable target given approximate evolution models and uncertain measurements. Parametrized shape-space, implicit probabilistic,

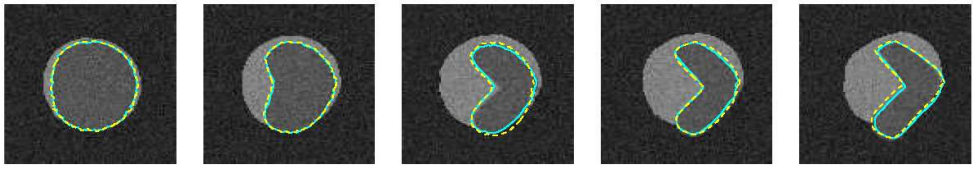


Fig. 16. Tracking through two non-affine contour modes of the likelihood and avoiding distractions by false edges [32].

and curve-based estimation models were reviewed, in addition to posterior mode particle filtering strategies. Except for the shape-based models, the methods detailed derive estimation strategies for curve measurements that are unconstrained, allowing for large contour deformations.

REFERENCES

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

[2] R. Brockett and A. Blake. Estimating the shape of a moving contour. In *Proceedings of IEEE Conference on Decision and Control*, pages 3247–3251, 1994.

[3] G. Chesi and K. Hashimoto, editors. *Visual Servoing via Advanced Numerical Methods*. Springer, 2010.

[4] D. Cremers. Dynamical statistical shape priors for level set based tracking. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(8):1262–1273, August 2006.

[5] D. Cremers. Nonlinear dynamical shape priors for level set segmentation. *Journal of Scientific Computing*, 35(2-3):132–143, June 2008.

[6] D. Cremers, D. Rousson, and R. Deriche. A review of statistical approaches to level set segmentation: Integrating color, texture, motion and shape. *International Journal of Computer Vision*, 72(2):195–215, 2007.

[7] D. Cremers, F. R. Schmidt, and F. Barthel. Shape priors in variational image segmentation: Convexity, lipschitz continuity and globally optimal solutions. In *Proceedings IEEE Conference on Vision and Pattern Recognition*, Anchorage, Alaska, June 2008.

[8] J. Cui, N. Ray, S.T. Acton, and Z. Lin. An affine invariance approach to cell tracking. *Computerized Medical Imaging and Graphics*, 32:554–565, 2008.

[9] S. Dambreville, Y. Rathi, and A. Tannenbaum. Tracking deformable objects with unscented Kalman filtering and geometric active contours. In *Proceedings of IEEE American Control Conference*, pages 1–6, 2006.

[10] S. Haker, G. Sapiro, A. Tannenbaum, and D. Washburn. Missile tracking using knowledge-based adaptive thresholding: Tracking of high speed projectiles. In *Proceedings of the IEEE International Conference on Image Processing*, pages 786–780, 2001.

[11] B. Horn and B. Schunck. Determining optical flow. *Artificial Intelligence*, 17:185–203, 1981.

[12] J.D. Jackson, A.J. Yezzi, and S. Soatto. Tracking deformable moving objects under severe occlusions. In *CDC*, pages 2990–2995, 2004.

[13] A. Mennucci, A. Yezzi, and G. Sundaramoorthi. Properties of Sobolev metrics in the space of curves. *Interfaces and Free Boundaries*, 10(4):423–445, 2008.

[14] P.W. Michor and D. Mumford. An overview of the Riemannian metrics on spaces of curves using the Hamiltonian approach. *Applied and Computational Harmonic Analysis*, 23:76–113, 2007.

[15] I.J. Ndiour, O. Arif, J. Teizer, and P.A. Vela. A probabilistic observer for visual tracking. In *Proceedings of IEEE American Control Conference*, 2010.

[16] I.J. Ndiour and P.A. Vela. Towards a local Kalman filter for visual tracking. In *Proceedings of IEEE Conference on Decision and Control*, pages 2420–2426, 2009.

[17] I.J. Ndiour and P.A. Vela. A local sub-optimal second-order filter for visual tracking. In *Proceedings of IEEE Conference on Decision and Control*, 2010.

[18] I.J. Ndiour and P.A. Vela. Optimal estimation applied to visual contour tracking. In *Proceedings of IEEE American Control Conference*, 2010.

[19] M. Niethammer, A. Tannenbaum, and S. Angenent. Dynamic active contours for visual tracking. *IEEE Transactions on Automatic Control*, 51(4):562–579, 2006.

[20] M. Niethammer, P.A. Vela, and A. Tannenbaum. Geometric observers for dynamically evolving curves. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(6):1093–1108, 2008.

[21] N. Papadakis and E. Mémin. A variational technique for time consistent tracking of curves and motion. *Journal of Mathematical Imaging and Vision*, 31(1):81–103.

[22] N. Peterfreund. Robust tracking of position and velocity with Kalman snakes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(6):564–569, 1999.

[23] G. Pryor, T. ur Rehman, S. Lankton, P.A. Vela, and A. Tannenbaum. Fast optimal mass transport for dynamic active contour tracking on the gpu. In *Proceedings of IEEE Conference on Decision and Control*, pages 2681–2688, 2007.

[24] Y. Rathi, S. Dambreville, M. Niethammer, J. Malcolm, J. Levitt, M. Shenton, and A. Tannenbaum. Segmenting images analytically in shape space. In *SPIE Symposium on Medical Imaging*, 2008.

[25] Y. Rathi, N. Vaswani, A. Tannenbaum, and A. Yezzi. Tracking deforming objects using particle filtering for geometric active contours. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(8):1470–1475, 2007.

[26] F. R. Schmidt and D. Cremers. A closed-form solution for image sequence segmentation with dynamical shape priors. In *Symposium of the German Associated for Pattern Recognition (DAGM)*, Jena, Germany, September 2009.

[27] L. Sigal, A.O. Balan, and M.J. Black. Humaneva: Synchronized video and motion capture dataset and baseline algorithm for evaluation of articulated human motion. *International Journal of Computer Vision*, 87(1-2):4–27, 2010.

[28] G. Sundaramoorthi, A. Menucci, S. Soatto, and A. Yezzi. Tracking deforming objects by filtering and prediction in the space of curves. In *Proceedings of IEEE Conference on Decision and Control*, 2009.

[29] G. Sundaramoorthi, A. J. Yezzi, and A. Mennucci. Coarse-to-fine segmentation and tracking using Sobolev active contours. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(5):851–864, 2008.

[30] P.H.S. Torr, R. Szeliski, and P. Anandan. An integrated Bayesian approach to layer extraction from image sequences. *TPAMI*, 23:297–303, 2001.

[31] A. Tsai, A. Yezzi, W. Wells, C. Tempany, A. Fan D. Tucker, E. Grimson, and A. Willsky. A shape-based approach to curve evolution for segmentation of medical imagery. *IEEE Transactions on Medical Imaging*, 22(2):137–154, 2003.

[32] N. Vaswani, Y. Rathi, A. Yezzi, and A. Tannenaum. Deform PF-MT: Particle filter with mode tracker for tracking non-affine contour deformations. *to appear IEEE Transactions on Image Processing*, 2010.

[33] P.A. Vela, M. Niethammer, G.D. Pryor, A.R. Tannenbaum, R. Butts, and D. Washburn. Knowledge-based segmentation for tracking through deep turbulence. *IEEE Transactions on Control Systems Technology*, 16(3):469–474, 2008.

[34] J. Xiao and M. Shah. Motion layer extraction in the presence of occlusion using graph cuts. *TPAMI*, 27:1644–1659, 2005.

[35] A. Yilmaz, X. Li, and M. Shah. Contour-based object tracking with occlusion handling. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(11):1531–1536, 2004.