SIAM J. IMAGING SCIENCES
Vol. 3, No. 4, pp. 835–855

A Probabilistic Contour Observer for Online Visual Tracking*

Ibrahima J. Ndiour†, Jochen Teizer‡, and Patricio A. Vela†

Abstract. This paper presents an online, recursive filtering strategy for contour-based tracking. Approaching the tracking problem from an estimation perspective leads to an observer design for the visual track signal associated with an individual target in an image sequence. The track state of the observer is decomposed into group and shape components that describe the gross location and the nonrigid shape, respectively, of the object. A probabilistic representation describes the shape nonparametrically. The constitutive components of the observer are detailed, which include a dynamical prediction model and a correction mechanism. Incorporating the probabilistic observer into the tracking process leads to improved performance and segmentations. The improvements are validated through application of the observer to recorded imagery with evaluation via objective measures of quality.

Key words. visual tracking, contour tracking, observer, shape, filtering

AMS subject classifications. 68U10, 93B07

DOI. 10.1137/100786629

1. Introduction.

This paper seeks to improve the online tracking of deformable moving objects in a video sequence through the use of an observer. The problem considered is that of accurate contour-based object tracking in the face of uncertainty caused by imaging noise and approximate segmentation models. Here, an object is a portion of a scene, captured by a visual sensor, with homogeneous properties (e.g., consistent appearance information such as intensity, color, or texture) that differentiate it from the rest of the scene. Online tracking captures the evolving object by segmenting the individual frames extracted from a video sequence. In its simplest instantiation, tracking consists of a series of statically determined contour measurements for each frame. This static approach yields good results so long as the target is sufficiently differentiated from the background. Poor differentiation arises from approximate or inadequate segmentation models, imaging noise, or occlusions. In those cases, this scheme may lose track of the target or result in degraded performance.

Imposing temporal consistency on the measurements is one strategy for managing imperfect or noisy measurements. To impose temporal consistency, [29, 43] process the image sequence in a volumetric fashion; they consider the entire video sequence at once, or several frames before and after the current frame, and solve a minimization problem whose solution guarantees temporal consistency and fitness to the segmentation model. The works in [39, 42]

*Received by the editors February 22, 2010; accepted for publication (in revised form) July 12, 2010; published electronically October 28, 2010. This work was funded by grants from the National Science Foundation under awards ECCS 0622006 and CMMI 0800858.
http://www.siam.org/journals/siims/3-4/78662.html
†School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA 30332-0250 (ndiour@gatech.edu, pvela@gatech.edu).
‡Department of Civil and Environmental Engineering, Georgia Institute of Technology, Atlanta, GA 30332-0355 (teizer@gatech.edu).

optimizes in batch a parametrized model for the target and the scene. While such processing techniques successfully impose temporal consistency of the solution, they operate in a context where access to future measurements is allowed, or measurement delay is tolerated. This work considers tracking techniques that operate in a recursive setting. Furthermore, the specific focus is on methods that produce both a trackpoint and a contour encircling the target.

1.1. Prior related work. Contour-based approaches for online tracking typically consider objects to be regions bounded by closed, planar curves [4, 19]. The space of closed, planar curves forms an infinite-dimensional manifold [20]. In recent years, attention has been devoted to the geometry of the manifold of curves [25, 44] and, in particular, to geodesics of closed curves [6, 20]. Closed curve geodesics have utility within the contexts of shape comparison and shape analysis.

The incorporation of shape constraints or of shape consistency benefits the tracking procedure. Many techniques that are robust to image-based disturbances utilize a priori shape information [10, 21]. A collection of shapes assumed to represent the shape space is typically analyzed using low-dimensional representation methods, e.g., principal component analysis (PCA) or kernel PCA (kPCA), yielding a finite-dimensional approximation for the actual infinite-dimensional shape space. The finite-dimensional representations factor into the segmentation algorithms so as to constrain the measurements. Extensions to shape-based methods include temporal modeling of the low-dimensional learned shape space [8, 9, 11]. Superb results are obtained using such techniques when the approximated low-dimensional space is suited to the video sequence at hand [11]. However, while it is plausible to assume sufficient prior knowledge of the target shape and movement for tracking certain sequences (rigid-body objects, or cyclic shape deformation for a walking person from a single viewpoint), for many scenarios it is unrealistic to make any assumption about the geometry and movement of the target. In an unconstrained setting, the target is able to undergo any feasible deformation of its shape, which the tracker should accurately recover.

The online tracking problem can be viewed as an estimation problem given temporally correlated measurements. A Markov assumption simplifies the problem to one of recursive estimation, for which one solution involves the use of an observer [13]. An observer reconciles a prediction of the current target state and a measurement of the current target state through a correction mechanism. Typically, the measurements are directly obtained from a sensor. In visual tracking, rather than coming from a sensor, the measurements are considered to be the output of a tracking algorithm.

Observers have been previously used to design visual tracking systems [12, 18, 26, 33, 38, 41]. The reference [45] defined a minimization strategy for estimating the shape and group motion associated with a deforming object given two or more segmentations of the object. The decomposition of an object's motion into a group motion and a shape deformation is not unique. However, the minimization provides a mechanism for consistently determining the decomposition over several frames and laying the mathematical foundation for a filtering strategy. The follow-up work [18] defined a correction strategy using the minimization algorithm to arrive at a joint group-shape space filter. In [12], PCA is used to reduce the shape space, whereby unscented Kalman filtering is performed on the resulting finite-dimensional space. The reduced finite-dimensional shape space is of great benefit: filtering strategies on finite-dimensional spaces (such as Kalman, extended Kalman, unscented Kalman, parti-

cle, etc.) may be used to filter the shapes [33, 41]. The drawback of such finite-dimensional shape approximations is related to the training step. In most cases, training requires carefully choosing the training set, performing the reduction analysis, and possibly learning the state evolution model in the reduced space, while facing the out-of-sample problem.

In contrast, the method in [28] derives an observer on the full infinite-dimensional manifold. The observer state implicitly encodes for the bounding contour through a signed distance function. However, on account of the state choice and measurement strategy, the observer has difficulty capturing rigid-body motion of objects. These limitations are overcome in [26, 38].

1.2. Contributions of this paper. This paper considers the problem of accurate trackpoint and boundary tracking of an unconstrained object given an image sequence. The hypothesis of the paper is that the appropriate construction of a dynamic observer for the track object's state, whose shape is determined by an implicit probabilistic contour, will lead to improved tracking performance. The observer provides a mechanism for the temporal history to influence the final segmentation of a given frame. With regards to the observer, object dynamics are decomposed into a group component and a shape component, with a subsequent decoupling of filtering strategies. A consequence of the observer is improved segmentations with more coherent and smooth target tracking.

Principal contributions include (1) the formulation of the tracking problem as an observer design problem on the group and shape, (2) the incorporation of a dynamical model for the probabilistic shape space, (3) the definition of a novel correction method suited to the probabilistic shape space description, and (4) the quantitative validation of the system's performance. These contributions are discussed within the context of existing work in the following paragraphs.

Instead of performing a joint minimization procedure, as done in [18], we decouple the filtering strategies on each component of the state, group, and shape, yielding filtered contour and trackpoint signals that combine to form the trackstate. A major limitation of the tracking system presented in [18] is the proposed joint minimization procedure and the choice of shape averaging as the correction method, which restricts potential targets to rigid or slowly deformable objects. Moreover, the observer components are fully embedded and described by the internal model. The work presented here derives a more modular structure by recasting the tracking problem as an estimation problem broken down into observer components for the group and shape substates. Exploiting the decomposition leads to the ability to define second-order motion models incorporating dynamics. The dynamic model is useful for capturing rapid or large deformations in shape. Moving away from implicit level-set curve representations, the paper utilizes a probabilistic model for the shape space, which better retains information about the classification of a given pixel through time than do level-set models where such information is determined by the curve boundary only (e.g., the zero level-set). In online tracking, relying on past information has practical utility. Thus, by using a pixelwise probabilistic description for the internal state, overall performance is improved. Finally, we present a correction strategy adapted to the shape space description and motivate its use for tracking.

We believe there are several benefits associated with this approach. The observer design is agnostic of the measurement strategy; i.e., any segmentation algorithm can be used to generate the measurements. With respect to the measurements, this paper presents a mechanism

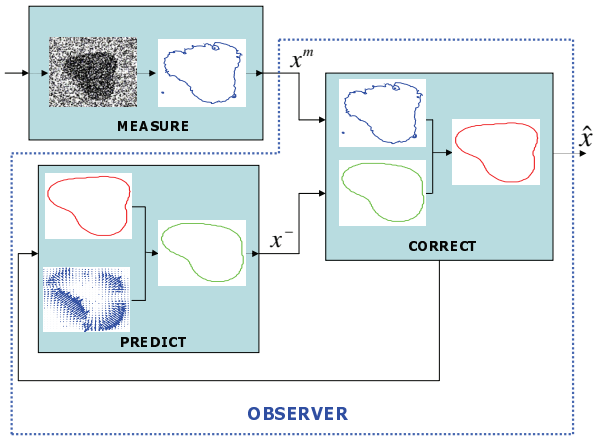


Figure 1. *Structure of an observer for a visual tracking system.*

to augment existing segmentation-based tracking techniques rather than to replace them. Secondly, as opposed to techniques using low-dimensional representations for the shape space, we examine temporal consistency in a nonparametrized setting. In practice, this translates into the proposed strategy's not requiring training steps and being immune to the out-of-sample problem, while still enforcing temporal consistency. Finally, defining robustness as the ability to track with minimal deviation from the truth, the approach presented here exhibits increased robustness. In particular, the observer enables accurate contour tracking that results in successful tracking over extended periods of time.

1.3. Organization of the paper. The paper is organized as follows. Sections 2 and 3 respectively describe the general structure of the observer and the observer components. A demonstration of the tracking improvement is given in section 4. Section 5 follows with a short conclusion and discussion of future work.

2. Observer setup for visual tracking. In control theory, *observers* or *estimators* are used to reconstruct the state of a system from (possibly incomplete) measurements [13]. In addition to estimating unmeasured state variables, an observer is capable of filtering noisy measurements. Within the context of visual tracking, the proposed observer will serve to both estimate unmeasured state quantities and filter noisy measurements. The structure of the proposed observer for visual tracking is given in Figure 1. It reflects the fact that filtering will be performed on the output of a visual tracking strategy rather than on the raw visual sensor data. Thus a state space must be defined for the internal states of the observer.

State description. The state of a deformable moving object comprises a group motion (pose) and a shape component describing the rigid motion and the nonrigid deformations of the object, respectively [45]. The group motion can be described by $SE(2)$ or its subgroup $E(2)$ [16]; the special Euclidean group in two dimensions, $SE(2)$, is a Lie group formed by the set of rigid-body motions in the plane. Its subgroup $E(2)$, the Euclidean group in two dimensions, is the set of translations in the plane. The nonrigid or shape component is given by a closed curve, meaning that the shape space is the space of closed curves. The closed curve description serves to partition the image domain into target and background regions.

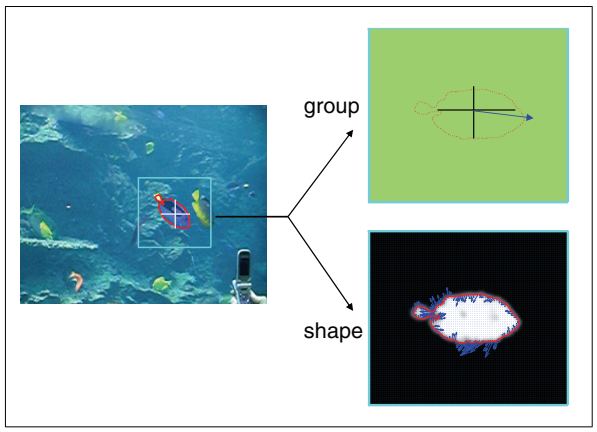


Figure 2. *Measurements of the group and shape. The description of a moving object is given by a group component that encodes for gross movement in the image frame and a shape component that encodes for non-rigid movement. To the left, the segmented object is displayed within the image, with the group component determining the origin of the velocity vector (top right) and the coordinate axis of the implicit shape description (bottom right). The shape description depicts both the current shape and the shape velocities.*

A standard representation for the closed curve $\mathcal{C}$ is achieved through its embedding into a higher-dimensional space, e.g., as the level-set of a signed distance function [35]. In practice, any function capable of implicitly describing a shape through the selection of an iso-contour will suffice. This work proposes a probability field defined over the image domain, $P : D \to [0, 1]$, where $D \subset \mathbb{R}^2$ compact. The implicitly defined contour $\mathcal{C}$ is recovered through the set $\mathcal{C} = \{r \in D \mid P(r) = \rho\}$, where $\rho \in (0, 1)$. Pixels with probabilities higher than or equal to ρ are presumed to belong to the target, and those with lower probabilities are presumed to belong to the background (we use the value $\rho = 0.5$). Consequently, the shape space for the work described herein is the space of probability fields defined over the image domain, an element of which will be denoted by P. We will also require the velocity field associated with P, here denoted by Θ. Thus, the state of the tracking system is described by the group variable g and the probability field P in conjunction with their associated velocities (g, P, ξ, Θ); see Figure 2.

3. Observer components. Observer implementation requires the definition of a prediction model for the state and a correction strategy for the internal model, given state measurements. State measurements serve as inputs to the observer, while the complete estimated state of the system is typically the output. The correction model is crucial to estimation performance: from a prediction and a measurement, it generates an updated estimate of the internal state of the observer. For the finite-dimensional group component, the correction is relatively straightforward. In comparison, there is not a unique method for performing correction on the shape component, due to the manifold nature of the space of closed curves.

Consider a linear finite-dimensional system in state space representation. In what follows, let the subscripts and superscripts m, $-$, and $+$ respectively denote measurement, prediction, and correction. The update equation of a finite-dimensional observer, with discrete correction

Table 1
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)	$\begin{cases} \dot{g} = \xi, \quad \dot{\xi} = 0 \\ \dot{P} + \nabla P \cdot \Theta = 0, \quad \dot{\Theta} = 0 \end{cases}$
Constant velocity (2)	$\begin{cases} \dot{g} = \xi, \quad \dot{\xi} = 0 \\ \dot{P} + \nabla P \cdot \Theta = 0, \quad \dot{\Theta} + \nabla \Theta \cdot \Theta = 0 \end{cases}$

times, is given by

$$\begin{bmatrix} x \\ \dot{x} \end{bmatrix}^{+} = \begin{bmatrix} x \\ \dot{x} \end{bmatrix}^{-} + \begin{pmatrix} K_{11} & K_{12} \\ K_{21} & K_{22} \end{pmatrix} \cdot \begin{bmatrix} err(x_m, x^{-}) \\ err(\dot{x}_m, \dot{x}^{-}) \end{bmatrix},$$

where $err(x_m, x^{-})$ and $err(\dot{x}_m, \dot{x}^{-})$ are the residuals in "position" and "velocity." The gain matrix $K = [K_{ij}]$ for $i, j = 1, 2$ is such that all entries lie in the range $[0, 1]$.

In the present case, an ideal correction scheme on the shape of the target would be to compensate the predicted shape by using the residuals in position and velocity, appropriately weighted with a gain matrix K. Because of the nature of the shape space and its probabilistic model, this scheme is infeasible. We propose instead a correction method adapted to the probabilistic shape description. The following sections further detail the observer inputs and components with regards to the visual tracking problem defined in the introduction.

3.1. Prediction model. The prediction model uses the state estimate from the previous frame to produce an estimate at the current frame. It is derived from a priori knowledge about the target evolution. With strong priors about the state dynamics, this framework allows the use of a detailed motion model to predict state estimates at future times from present and past state estimates. In some cases, generic motion models exist that can sufficiently capture the system dynamics. For more specific motion models, one can appeal to available literature or otherwise derive a motion model from first principles.

In this paper, we propose two general purpose dynamic prediction models for the shape probability field. The motion models are summarized in Table 1, along with the simpler general models currently found in the literature (static and constant group velocity). The constant velocity models in Table 1 push forward the probability field according to estimated velocities of the target domain. The second model is differentiated from the first through the advection of the velocity field with the probabilities. The incorporation of second-order dynamics will more effectively predict the motion of deforming targets than will static shape models, such as the static prior and the constant group velocity prior (as used in [45]). The transport equations for the shape and associated velocities given in Table 1 are simple partial differential equations involving the partial derivatives in time and space, respectively denoted as $\dot{X}$ and ∇X when applied to the variable X. Real-time implementations for the motion models exist [37].

3.2. Measurement. As depicted in Figure 1, the measurement block is not part of the observer even though it is an important component of the visual tracking system. The state measurements occur external to the observer and provide it with the input. The measurement module involves, at most, the determination of the four substates associated with the tracker, $(g_m, \xi_m, P_m, \Theta_m)$. In practice, the group velocity ξ_m is not normally available for measurement, as it is not directly measurable from the image, and must be estimated. While Θ can be approximately measured from the image sequence, doing so is optional.

3.2.1. Group/probability field. Measurement of the target shape can be achieved through any segmentation algorithm applied to the current image, so long as the segmentation is converted to the implicit probability field description. Candidate algorithms include Bayesian segmentation [15], active contours [5], graph cuts [22], etc.

In a classical observer the measurements would be completely independent of the observer states; however, image analysis of video sequences has the unique nature of not explicitly providing the necessary signal. Instead such a signal must be extracted from the image using an image processing or computer vision algorithm. The measurement procedure may not completely determine the necessary target state measurements (due to the nonuniqueness of the group plus shape decomposition). Consequently, a registration step is required to describe the predicted and measured shape with respect to the same coordinate frame.

Once localization [7, 31, 36] 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 a measurement P_m for the shape. Registration can be achieved through the use of deformotion to identify the group transformation only [45], or through another suitable algorithm.

3.2.2. Velocity field. The velocity field Θ_m can be measured by computing the optical flow [17] between two subsequent aligned images I_{n-1} and I_n. Since direct implementation of the optical flow field might not allow the capture of large displacements, one may need to refine the algorithm in order to fully capture the field [1]. One example of such refinement would be to consider pyramidal implementations. Alternatively, rather than using the velocity field traditionally computed by the optical flow, one can use another variant based on the displacement field. The displacement field minimizes the energy functional

$$E(u,v) = \int_D \left[I_n(x,y) - I_{n-1}(x+u, y+v)\right]^2 dx\, dy \\ + \lambda \cdot \int_D \left[||\nabla u(x,y)||^2 + ||\nabla v(x,y)||^2 \right] dx\, dy.$$

The minimization of the functional can be performed through a simple iterative Gauss–Seidel procedure:

$$\begin{cases} u_{\tau+1} = u_\tau + d\tau \cdot \left(\lambda \Delta u + I_x \cdot [I_n - I_{n-1}(x+u, y+v)]\right), \\ v_{\tau+1} = v_\tau + d\tau \cdot \left(\lambda \Delta v + I_y \cdot [I_n - I_{n-1}(x+u, y+v)]\right), \end{cases}$$

where Δ denotes the Laplacian operator, τ is an artificial time parameter, $d\tau$ is the time step, (u_τ, v_τ) represents the displacement field at time τ, and (I_x, I_y) is the gradient in space of the image I_{n-1}. The resulting error vector field $\Theta_{err} = (u, v)$ is able to capture large disparities between the images I_{n-1} and I_n.

Table 2
State correction model. (EKF: extended Kalman filtering; UKF: unscented Kalman filtering)

State component	Correction description
Group and group velocity	Correction according to the update equations of the finite-dimensional filter adopted (Kalman, EKF, or UKF).
Shape	$\widehat{P}^{+} = (\widehat{P}^{-})^{1-K_{11}} \cdot (P_m)^{K_{11}}$
Shape velocity	$\widehat{\Theta}^{+} = \widehat{\Theta}^{-} + K_{21} \cdot \Theta_{err}(P_m, \widehat{P}^{-}) + K_{22} \cdot (\Theta_m - \widehat{\Theta}^{-})$

3.2.3. Regarding the shape measurement. Recall that the observer is agnostic to the measurement strategy. If the segmentation algorithm does not automatically generate a probability field, then conversion to implicit probability field form is required. For example, consider an active contour implicitly represented by a signed-distance function, $\Psi : D \to \mathbb{R}$. Conversion to a probabilistic description is realized using the regularized Heaviside function [5], whose values lie in the range $[0, 1]$,

$$P(\cdot) = \frac{1}{2}\left(1 + \frac{2}{\pi}\arctan\left(\frac{\Psi(\cdot)}{\sigma_r}\right)\right),$$

or by applying the cumulative density function of the normal distribution (with zero mean) to the negative signed-distance function,

$$P(\cdot) = \mathrm{cdf}(-\Psi(\cdot); \sigma_r) = \frac{1}{2}\left(1 + \mathrm{erf}\left(\frac{\Psi(\cdot)}{\sigma_r\sqrt{2}}\right)\right),$$

where σ_r is the standard deviation and erf denotes the Gauss error function defined by $\mathrm{erf}(x) = \frac{2}{\sqrt{\pi}}\int_0^x e^{-t^2}\,dt$. In both cases, σ_r is a regularization parameter.

3.3. Correction. Given state measurements, the correction model refines the current estimate of the state. Due to the nonuniqueness of the group and shape decomposition, the measurement procedure may not properly determine the necessary target state measurements for comparison against the predicted state. A registration step is required to describe the predicted and measured shapes with respect to the same coordinate frame [45]. Thus, once localization and segmentation is performed on the current image, a registration procedure is applied to match the resulting measured probability field with the predicted probability field. The registration procedure yields the group error, which is also the necessary transformation to describe the two shapes within the same coordinate system. Table 2 summarizes the correction model as detailed in the remainder of this section.

3.3.1. Group. In this work, the group motion is described by $SE(2)$, or its subgroup $E(2)$. It is given by a translation in the x-axis, a translation in the y-axis, and possibly a rotation of the axes. Correction on the group motion can then be done through classic filtering using a three-state vector (appropriately augmented). If a linear motion model such as the one presented in Table 1 is used, Kalman filtering is recommended. When the motion model is believed to be nonlinear, nonlinear filtering methods such as extended or unscented Kalman filtering should be considered. Detailed information about Kalman filtering can be found in [24].

3.3.2. Shape. This paragraph motivates the use of geometric averaging for the correction scheme through an analysis of the effect of additive noise on the shape density when using Bayesian classification to identify foreground and background regions.

Setup. For simplicity, consider a scalar valued image $I : D \to \mathbb{R}$ defined over $D \subset \mathbb{R}^2$ compact. Further, assume that measurement of the pixel intensities has been corrupted by additive Gaussian noise ν with zero mean and variance σ_ν^2. Segmentation can be performed through Bayesian classification [15] 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

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

where c is a positive normalizing factor. The expression for the measured likelihood can be expanded further to give

$$\zeta_F(r) = \sqrt{c} \cdot e^{-\frac{1}{2}\left(\frac{I(r)-\mu_F}{\sigma_F}\right)^2} \cdot e^{-\frac{1}{2}\left(\frac{\nu}{\sigma_F}\right)^2} \cdot e^{-\left(\frac{\nu\,(I(r)-\mu_F)}{\sigma_F^2}\right)}, \tag{3.1}$$

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 (3.1) is the true classification likelihood, and $\eta(r; \mu_F, \sigma_F)$ consisting of the remaining terms is the class measurement noise. Thus, corruption by additive noise on the image data results in multiplicative uncertainty for the foreground likelihood. The multiplicative uncertainty is correlated with the image distribution. A similar derivation holds for the background classification densities. Further, the extension to vector valued imagery follows naturally.

Estimation with multiplicative noise. In the log-space associated with the probabilities,

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

the uncertainty associated with the additive image noise becomes additive (but nonlinear). Now, suppose that the current estimate for the foreground probability field is $\widehat{P}_F^-$, while the current measurement of the probability field is ζ_F. Applying pointwise a constant gain linear filtering strategy to filter the noise leads to the corrected estimate of the probability field,

$$\log(\widehat{P}_F^+) = \log(\widehat{P}_F^-) + K\,\left[\log(\zeta_F) - \log(\widehat{P}^-)\right].$$

Rearranging the terms, we obtain

$$\log(\widehat{P}_F^+) = (1-K)\log(\widehat{P}_F^-) + K\log(\zeta_F).$$

Returning to the density space by applying the exponential gives

$$\widehat{P}_F^+ = \left(\widehat{P}_F^-\right)^{1-K} (\zeta_F)^K.$$

Thus, a suboptimal filtering strategy for managing uncertainty is to perform geometric averaging of the segmentation probabilities. The filtering applies to both the foreground and background probability fields.

Application to visual tracking. Correction on the shape will be performed using geometric averaging. It was just shown that, under the proper assumptions, the use of probabilities for the measurement strategy induces multiplicative uncertainty. In the log-space, where multiplicative error becomes additive error, linear correction is performed. After correction in the log-space, transformation back to the probability space results in geometric averaging.

The geometric averaging correction strategy for the shape probability field is applied pointwise,

$$\widehat{P}^{+}(r) = \left(\widehat{P}^{-}(r)\right)^{1-K_{11}(r)} \left(\widehat{P}_m(r)\right)^{K_{11}(r)} \qquad \forall r \in D.$$

The parameter K_{11} is defined by the user according to the measurement noise. Low K_{11} is for high measurement noise, since the correction is biased towards the prediction. High K_{11} is for low measurement noise, since the correction is biased towards the measurement. Here, the gain K_{11} is constant temporally but may vary spatially. For example, since the shape is always centered, low gain values may be applied near the image boundaries to forbid shape update where the target is known to be absent. Last, the gain K_{12} is set to zero due to the structure of the shape space.

3.3.3. Shape velocities. The shape velocities live in the tangent space to the shape probability field, meaning that shape velocities lie in the space of two-dimensional vector fields defined on D. Corrections on the velocity field occur through an error in the measured probability fields, Θ_{err}, and an error in the measured velocities:

$$\widehat{\Theta}^{+} = \widehat{\Theta}^{-} + K_{21} \cdot \Theta_{err}(P_m, \widehat{P}^{-}) + K_{22} \cdot (\Theta_m - \widehat{\Theta}^{-}).$$

The error vector field Θ_{err} is not uniquely defined and will vary according to the notion of geodesy, as discussed in the introduction. One method for generating the error Θ_{err} is to compute the optical flow between the measured and predicted probability fields, while another is to utilize optimal mass transport [32].

4. Experiments and results.

4.1. Setup. The observer-based tracking system was tested on a body of construction footage and aquarium sequences. The basic measurement strategy chosen was Bayesian segmentation [15]. In addition to tracking with Bayesian segmentation, several other methods were tested on the sequences. The third algorithm used is from [18], which is a joint filtering strategy for both the group and the shape, and is labelled Deformotion in Figure 3 and thereafter. An active contour technique [34] has also been applied based on the same distributions used by the Bayesian algorithm; it is labelled AC. A standard additional regularizing term is present in the active contour energy whose relative weight with respect to the probability competition terms has been manually tuned for each sequence to optimize performance. The parameter regulates convergence to the object (external energy) while penalizing irregularities of the contour (internal energy) [3]. The fifth comparison technique is a shape-based method derived from [40], similar to [9] and labelled Shape. The shape-based segmentation strategy uses the same distributions as the Bayesian algorithm. Often in the literature, the shape-based trackers utilize the learned shape priors from the same image sequence to be tracked. In more realistic scenarios, one cannot presume the existence of shape information

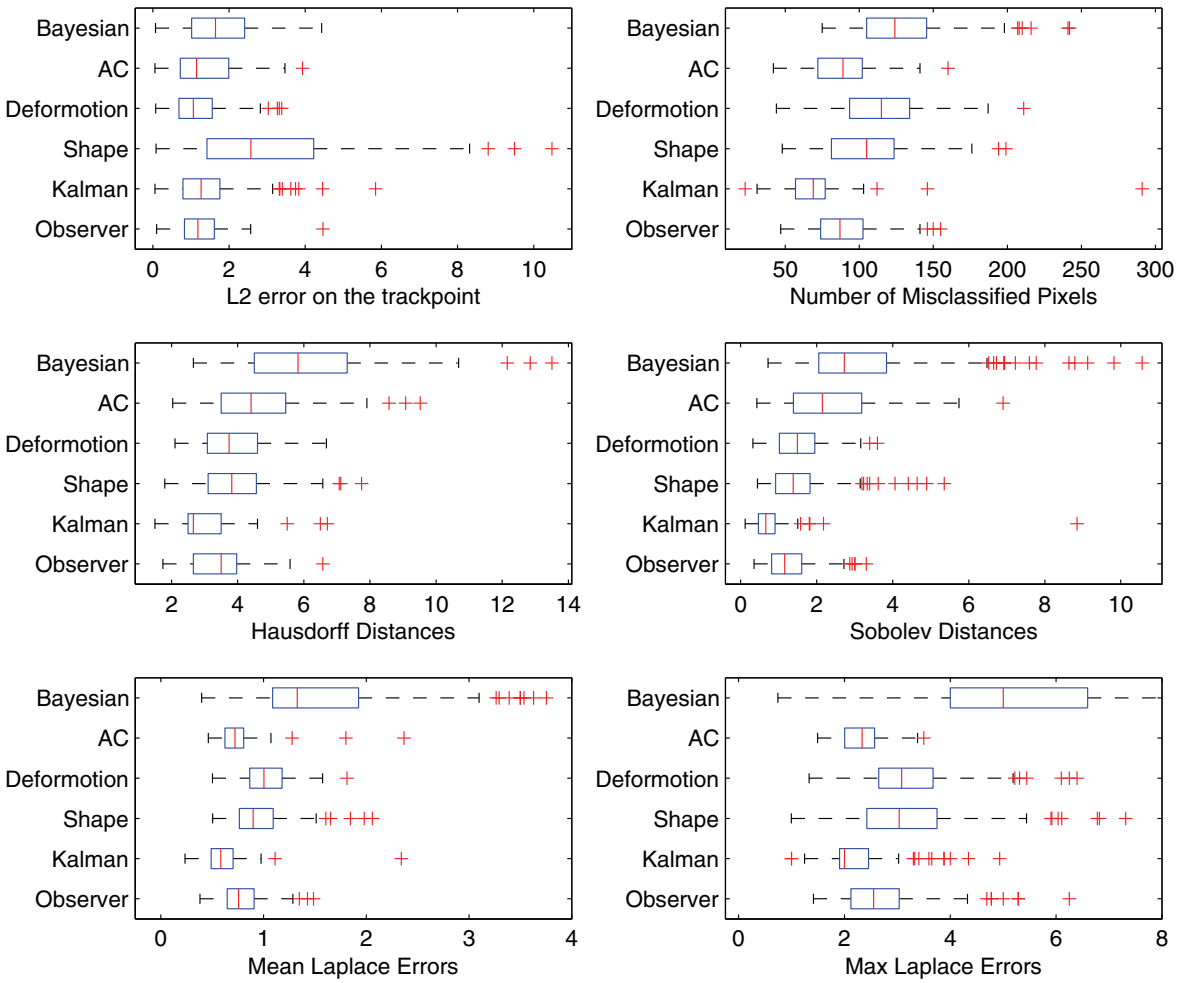


Figure 3. *Box plots of the comparative statistics on the infrared sequence. For each metric/algorithm pair, the corresponding box represents the error distribution, with lines at the lower quartile, median, and upper quartile values. The extent of the remaining data is displayed by the lines extending out from each box. Outliers are denoted by the + symbol.*

for all possible persons/objects to be tracked. Consequently, we obtained 67 sample shapes from our construction imagery database, and the ten dominant eigenmodes were kept. For the aquarium sequence, seven shapes were considered, and the five dominant eigenmodes were retained. For all methods except deformotion, we applied the same filter to the group space, while the shape-based method also utilized a Kalman filter on the finite-dimensional shape parameters.

For the noisy infrared sequence, we used a sixth comparison algorithm based on a finite-dimensional approximation of the shape space. From a collection of 30 shapes extracted from the infrared sequence, we used an autoregressive model [27] to learn the state evolution in the PCA-reduced shape space, as in [8]. At each frame, the Kalman tracker operates by projecting the shape obtained through the Bayesian algorithm on the PCA space; the resulting vector

is then applied to a Kalman filter and reconstructed back in the shape space for display and comparison purposes.

4.2. Validation. In order to validate the effectiveness of the observer, the output signal (group plus shape) was compared against a ground truth signal using a variety of error metrics. Manual segmentations of all sequences served as the ground truth. For the group signal, we used the L_2 and L_∞ errors with respect to the ground truth. For the shape signal, we used the number of misclassified pixels (NMP), the Hausdorff distance, and the Sobolev distance [6, 23]. For the infrared sequence, we also used the mean and maximum Laplace errors [30]. For more information about shape metrics, see [6, 46, 14, 20] and references therein. The performance summary tables (Tables 3–4) give the median value of the shape error computed for all frames successfully tracked by the given method, and also the maximum shape error throughout the sequence. The median value gives an indication of how well on average a given technique is performing, while the maximum value highlights the worst behavior.

The sample tracking snapshots (Figures 6–10 below) display three chosen frames for each tracking algorithm: the left frame depicts the lowest shape error, the middle frame illustrates a sample shape error at the median, and the right frame shows the largest shape error. Strikeouts in the text indicate a loss of track.

4.3. Comments. The first set of observations will be made on the results obtained for the noisy infrared sequence. The Bayesian measurement strategy results in the worst performance with relatively large deviations. The quality of the segmentations is shown in Figure 4.

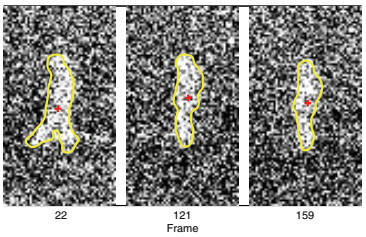
The active contour better manages the noise, on account of the regularization term, but has a tendency to undersegment. The deformotion tracking, through its shape averaging correction, attenuates the noise but also oversmooths, as reflected by the number of misclassified pixels and the Sobolev distance. Listing the algorithms as {Bayesian/AC, Deformotion, Shape, Kalman} orders them according to the quantity of prior information embedded into the algorithms (from least to most). It is therefore to be expected for the metrics to indicate a ranking of performances in that order. The Bayesian observer, lying between the deformotion and shape methods in term of information requirements, exhibits an ability to filter out the noise while still matching the segmented shapes to the ground truth. When compared to the other algorithms (Figure 4), the Bayesian observer has the best overall performance after the Kalman tracker. The Kalman tracker in this instance represents the best possible tracker. The dynamics of the trackstate have been fully analyzed and described in the dynamical model, plus the learned shapes were obtained directly from the image sequence. In most scenarios, one is unlikely to have available a tracker with such prior knowledge and with the guarantee that the parametrized shape space will perfectly describe the target movement.

The remainder of the discussion will revolve around the video sequences whose performance evaluation is depicted in Table 3. Sample frames are given in Figure 5. The table summarizes the performance results obtained using various performance measures. For each of these metrics and each algorithm, we provide the median value and the maximum value of the corresponding error throughout the sequence. When a given technique fails to track the target throughout the entire sequence, the corresponding column is stricken out.

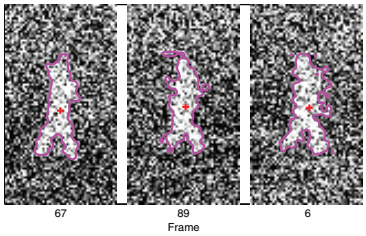
The first construction sequence (Figure 6) contains a worker that is bending at some point in the video (frame 57 to frame 162). Even though a couple of the bending segmentations were

(a) Sample from original sequence.

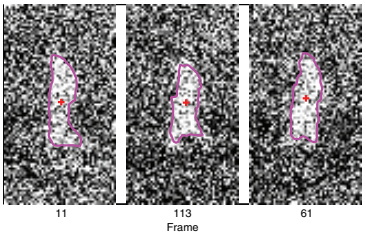
(b) Sample from noisy sequence.



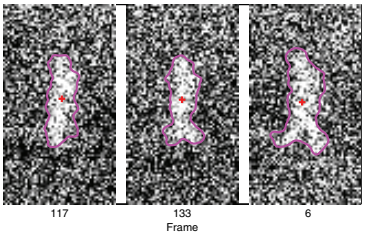
(c) Sample ground truth.



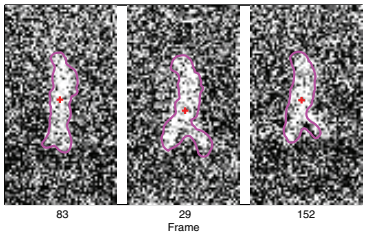
(d) Bayesian tracker.



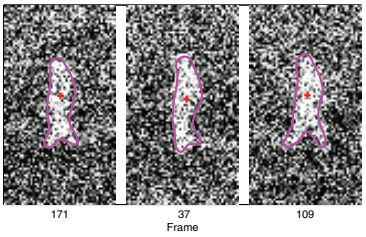
(e) Active contour.



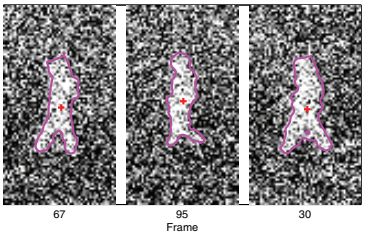
(f) Deformotion tracker.



(g) Shape-based.



(h) Kalman-based tracker.



(i) Observer-based tracker.

Metric \ Algorithm		Bayesian	AC	Deformotion	Shape	Kalman	Observer
Trackpt error	(L_2/L_∞)	1.8 / 4.4	1.4 / 3.9	1.2 / 3.4	3.0 / 10.5	1.4 / 5.8	1.2 / 4.4
NMP	(med - max)	129 / 242	91 / 160	116 / 211	105 / 199	70 / 146	90 / 155
Hausdorff	(med - max)	6.2 / 13.5	4.5 / 9.5	4.0 / 6.7	3.9 / 7.7	3.0 / 6.7	3.5 / 6.6
Sobolev	(med - max)	3.2 / 10.5	2.4 / 6.9	1.5 / 3.6	1.5 / 5.4	0.8 / 8.8	1.2 / 3.3
Mean Laplace	(med - max)	1.9 / 9.8	0.7 / 2.4	1.1 / 1.8	1.0 / 4.7	0.6 / 2.3	0.8 / 1.5
Max Laplace	(med - max)	5.5 / 13.0	2.3 / 3.5	3.2 / 6.4	3.3 / 8.0	2.2 / 4.9	2.8 / 6.2
# Frames tracked		180	180	180	180	180	180

(j) Table summarizing the statistics.

Figure 4. *Noisy infrared sequence. For each algorithm (in* (d)–(i)*), the left frame represents the best frame tracked, the middle frame shows how the technique operates on average, and the right frame displays the worst segmentation obtained. The table highlights the results obtained when comparing the different tracking techniques with the ground truth using objective measures of quality. For each of these metrics and each algorithm, we provide the median value and the maximum value of the corresponding error throughout the sequence.*

Table 3
Comparative performance of applied algorithms.

(a) Construction sequence 1.

Metric \ Algorithm	Bayesian	AC	Deformotion	Shape	Observer
Trackpt error (L_2/L_∞)	16.6 / 24.4	~~11.5 / 52.3~~	7.9 / 16.0	5.4 / 12.3	8.0 / 15.5
NMP (med/max)	253 / 1420	~~288 / 1328~~	202 / 755	299 / 536	171 / 508
Hausdorff (med/max)	10.2 / 35.0	~~30.0 / ∞~~	7.8 / 26.2	10.9 / 25.8	7.7 / 27.4
Sobolev (med/max)	8.2 / 70.6	~~100.0 / ∞~~	5.8 / 35.3	11.7 / 38.1	6.5 / 81.8
# Frames tracked	200	~~150~~	200	200	200

(b) Construction sequence 2.

Metric \ Algorithm	Bayesian	AC	Deformotion	Shape	Observer
Trackpt error (L_2/L_∞)	11.5 / 13.3	8.6 / 12.4	8.3 / 12.4	6.5 / 9.1	8.3 / 12.4
NMP (med/max)	84 / 183	98 / 194	94 / 146	264 / 354	81 / 132
Hausdorff (med/max)	3.9 / 7.5	5.7 / 12.0	4.7 / 9.2	8.9 / 13.2	3.9 / 8.1
Sobolev (med/max)	1.6 / 5.0	4.5 / 33.3	3.1 / 10.8	14.0 / 23.8	2.2 / 9.2
# Frames tracked	100	100	100	100	100

(c) Construction sequence 3.

Metric \ Algorithm	Bayesian	AC	Deformotion	Shape	Observer
Trackpt error (L_2/L_∞)	9.1 / 10.9	~~6.2 / 23.6~~	2.1 / 4.5	~~4.6 / 20.5~~	2.1 / 4.5
NMP (med/max)	131 / 315	~~206 / 451~~	113 / 213	~~158 / 282~~	114 / 219
Hausdorff (med/max)	6.5 / 14.4	~~10.5 / 23.0~~	5.0 / 9.8	~~6.1 / 12.5~~	5.0 / 12.0
Sobolev (med/max)	4.5 / 18.1	~~31.5 / 87.8~~	11.2 / 40.2	~~12.4 / 27.9~~	4.5 / 18.7
# Frames tracked	150	~~122~~	150	~~135~~	150

(d) Construction sequence 4.

Metric \ Algorithm	Bayesian	AC	Deformotion	Shape	Observer
Trackpt error (L_2/L_∞)	~~3.8 / 8.2~~	~~5.7 / 10.8~~	~~2.7 / 6.7~~	~~3.0 / 9.8~~	2.7 / 6.7
NMP (med/max)	~~51 / 143~~	~~40 / 103~~	~~39 / 129~~	~~93 / 167~~	38 / 188
Hausdorff (med/max)	~~6.8 / 16.5~~	~~3.7 / 27.3~~	~~3.7 / 13.7~~	~~9.2 / 19.1~~	4.6 / 14.3
Sobolev (med/max)	~~6.7 / 86.1~~	~~4.8 / 220.4~~	~~2.9 / 25.4~~	~~13.1 / 30.4~~	3.7 / 23.2
# Frames tracked	~~657~~	~~403~~	~~640~~	~~418~~	1010

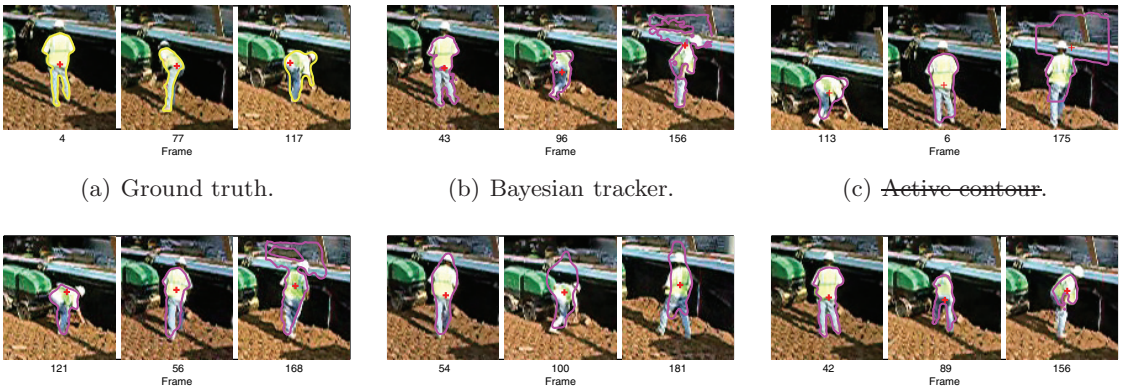
(e) Aquarium sequence.

Metric \ Algorithm	Bayesian	AC	Deformotion	Shape	Observer
Trackpt error (L_2/L_∞)	8.6 / 13.2	2.8 / 7.0	2.6 / 12.3	5.6 / 15.8	2.7 / 5.8
NMP (med/max)	251 / 969	244 / 549	248 / 769	575 / 833	279 / 478
Hausdorff (med/max)	10.9 / 18.4	11.1 / 19.2	12.3 / 19.7	12.0 / 22.5	14.6 / 20.7
Sobolev (med/max)	8.2 / 52.9	12.9 / 95.8	11.9 / 46.7	13.2 / 43.9	12.9 / 26.9
# Frames tracked	477	478	477	475	478

used at the training step, they did not factor into the main eigenmodes. Consequently, the shape-based technique is unable to find the correct shape for the portion of the video where the man is bending. It may be possible to improve the segmentation with the shape-based method

(a) Sequence 1. (b) Sequence 2. (c) Sequence 3. (d) Sequence 4.

Figure 5. *Sample frames from construction video sequences.*



(a) Ground truth. (b) Bayesian tracker. (c) ~~Active contour~~.

(d) Deformotion tracker. (e) Shape-based. (f) Observer-based tracker.

Figure 6. *Tracking snapshots of construction sequence* 1 *(strikeouts indicate loss of track).*

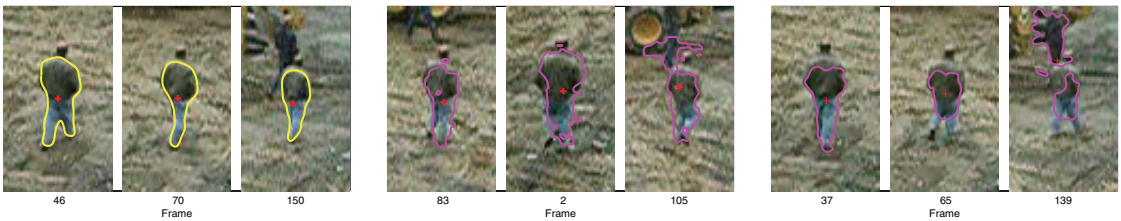
by allowing for more eigenmodes. However, to consider all possible shapes would increase the computational complexity of the technique and significantly increase the size of the training set. A further difficulty of the sequence is the fact that the target and background distributions overlap somewhat, which leads to some ambiguity regarding the proper segmentation. The depiction of the worst shape error indicates that all of the methods, with the exception of the Bayesian observer, were unable to reject the segmentation disturbance. By reducing the measurement information to the segmentation boundary, the deformotion method effectively ignores any temporal history associated with the interior target regions. The Bayesian observer maintains a history of the segmentation probabilities pixelwise and can thus more effectively reject spurious segmentation leakages that occur in individual frames. The worst-case scenario for the Bayesian observer indicates a more conservative estimation of the shape than for the other filter-based methods.

Sequence 2 (Figure 7) features a construction worker going down a slope. The Bayesian segmentation method performed well, as reflected by the quality metrics given in Table 3. This example illustrates that by applying the Bayesian observer, one can still improve the quality of the segmentations. Not only do we enforce temporal consistency, but thanks to the second-order model that incorporates dynamics, we are able to maintain a coarse definition of the contour around the legs of the man. In this case, due to shape averaging, the deformotion

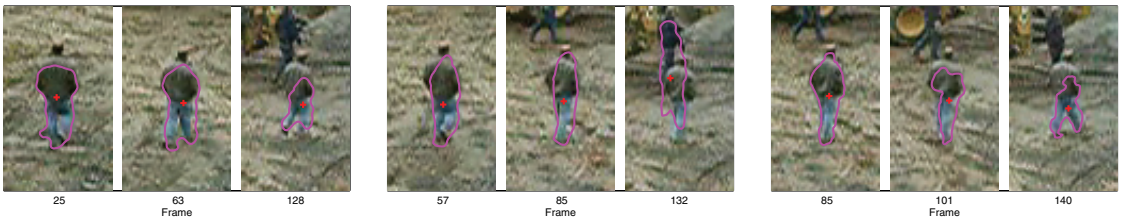
(a) Ground truth. (b) Bayesian tracker. (c) Active contour.

(d) Deformotion tracker. (e) Shape-based. (f) Observer-based tracker.

Figure 7. *Tracking snapshots of construction sequence* 2.



(a) Ground truth. (b) Bayesian tracker. (c) ~~Active contour~~.

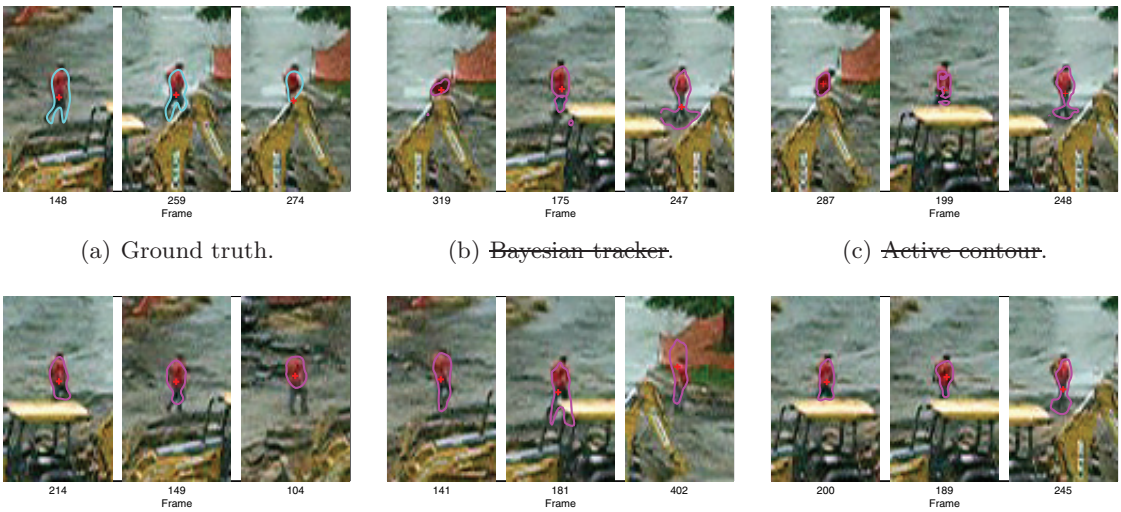


(d) Deformotion tracker. (e) ~~Shape-based~~. (f) Observer-based tracker.

Figure 8. *Snapshots of construction sequence* 3 *(strikeouts indicate loss of track).*

method consistently fails to capture the legs, as can be seen in the middle frame (Figure 7).

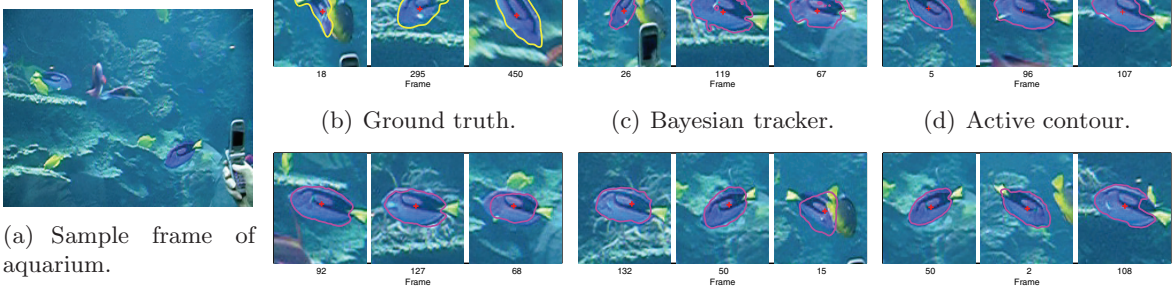
Sequence 3 (Figure 8) features a working man on a construction site who approaches another man with similar colors/distribution. Since the Bayesian tracker and the active contour obtain a trackpoint by computation of the segmentation centroid, whenever the segmentation fails and grabs portions of the background, their trackpoints deviate. Moreover, the trackpoints fluctuate due to their dependency on shape. Even though the Bayesian observer relies on the segmentation, it uses a registration procedure to decompose the trackstate into group and shape, and the resulting trackpoint is immune to segmentation fluctuations: the



(a) Ground truth. (b) ~~Bayesian tracker~~. (c) ~~Active contour~~.

(d) ~~Deformotion tracker~~. (e) ~~Shape-based~~. (f) Observer-based tracker.

Figure 9. *Snapshots of construction sequence 4 (strikeouts indicate loss of track).*



(a) Sample frame of aquarium.

(b) Ground truth. (c) Bayesian tracker. (d) Active contour.

(e) Deformotion tracker. (f) Shape-based. (g) Observer-based tracker.

Figure 10. *Sample frame and snapshots of the aquarium sequence.*

trackpoint remains anchored even in the presence of perturbations.

The last construction sequence (Figure 9) shows a worker walking around an excavator. Difficulties associated with the sequence include the small size of the target compared to the image dimensions, significant clutter, noise, and partial occlusion. The Bayesian observer proves its robustness by tracking the whole sequence (1010 frames) with good performance. By comparison, all other techniques fail to track up to frame #700 (Table 3).

The aquarium sequence (Figure 10) is an example of a moving object that undergoes large shape deformations during motion reversals. Again, there is some overlap of the target and background distributions. Visual inspection of the tracked videos show that the observer better enforces temporal consistency on the contour, with deformotion a close second. The results are confirmed by the performance metrics.

Table 4

Table of statistics for observer using measurements obtained from an active contour and graph cut segmentations.

(a) Noisy infrared sequence.

Metric \ Algorithm		Observer with AC measurements	Graph Cut	Observer with Graph Cut measurements
Trackpt error	(L_2/L_∞)	2.5 / 5.6	2.4 / 9.8	1.6 / 4.3
NMP	(med/max)	93 / 133	293 / 480	105 / 178
Hausdorff	(med/max)	4.2 / 8.7	14.1 / 16.5	4.2 / 7.9
Sobolev	(med/max)	1.6 / 4.4	23.0 / 37.6	1.7 / 6.3
Mean Laplace	(med/max)	0.9 / 4.6	9.2 / 16.1	0.9 / 2.0
Max Laplace	(med/max)	2.8 / 4.9	11.1 / 16.3	3.7 / 7.8
# Frames tracked		180	180	180

(b) Sequence 1.

Metric \ Algorithm		Observer with AC measurements	Graph Cut	Observer with Graph Cut measurements
Trackpt error	(L_2/L_∞)	6.5 / 22.1	7.9 / 31.1	6.9 / 27.4
NMP	(med/max)	192/ 663	288 / 1014	219 /457
Hausdorff	(med/max)	8.3 / 25.4	12.8 / 32.0	11.7/ 25.9
Sobolev	(med/max)	6.2 / 35.8	8.3 / 70.8	10.2/ 80.1
# Frames tracked		200	200	200

Application to other measurement methods. To demonstrate the fact that other segmentation-based tracking methods would benefit from the use of the probabilistic observer, the observer strategy was applied to tracking methods relying on active contour and graph-cut segmentation. Table 4 shows the results for two of the sample sequences. Results obtained with the segmentation methods (active contour [34] and graph cuts [2]) are improved by the observer. Originally, the active contour alone could not track the complete sequence (see Table 3(a)). With the addition of the observer, complete tracking was achieved.

The Bayesian observer compares favorably with the other techniques used here. In the face of a large variety of disturbances to the segmentation process, the observer preserves target track and improves temporal consistency of the track signal. Using a range of metrics, we have compared it objectively to several tracking techniques of the same family. The results of these investigations show that the observer results in equivalent performance in the absence of disturbances. When perturbation occurs, the observer eliminates or minimizes the resulting undesired segmentation effects. Even when it is not able to completely correct poor measurements, temporal consistency is maintained, and the perturbations manifest as conservative segmentations (Figures 7 and 9) rather than as a noisy shape. The technique has minimal computational complexity and can easily run at frame rate (in a MATLAB implementation, we can achieve 2–3 frames per second or more, depending on the target size).

5. Conclusion. This paper presented the design of an observer with a probabilistic shape representation for the purpose of improving tracking performance of segmentation-based track-

ing algorithms. As part of the observer, generic dynamic prediction models are given, and a local shape correction adapted to the implicit probabilistic representation is described. Experiments conducted on recorded color videos, together with quantitative performance metrics, demonstrate that the proposed observer improves both temporal consistency and tracking robustness.

The correction algorithm currently utilizes fixed gains. Future work will involve the derivation of an optimal gain strategy for the observer based on known uncertainty levels associated with the state estimate, the prediction, and the measurement. A further avenue of inquiry arising in the context of this study is the role of shape. Shape is necessary when there is a massive amount of false positive image content extending over long times. Current methods use hard shape constraints to restrict the measurement to lie within a low-dimensional parametrized space. We suspect that a more robust approach consists of adopting looser constraints on the measurement model while enforcing harder constraints on the correction model, in contrast to the prevailing methods which impose hard constraints during measurement.

REFERENCES

[1] S. Baker, D. Scharstein, J. P. Lewis, S. Roth, M. J. Black, and R. Szeliski, *A database and evaluation methodology for optical flow*, in Proceedings of the Eleventh IEEE International Conference on Computer Vision, Rio de Janeiro, Brazil, 2007, IEEE Press, Piscataway, NJ, pp. 1–8.

[2] Y. Boykov and V. Kolmogorov, *An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision*, IEEE Trans. Pattern Anal. Machine Intell., 26 (2004), pp. 1124–1137.

[3] V. Caselles, *Geometric models for active contours*, in Proceedings of the IEEE International Conference on Image Processing, Washington, DC, 1995, IEEE Press, Piscataway, NJ, pp. 9–12.

[4] V. Caselles, R. Kimmel, and G. Sapiro, *Geodesic active contours*, in Proceedings of the Fifth International Conference on Computer Vision, Cambridge, MA, 1995, IEEE Press, Piscataway, NJ, pp. 694–699.

[5] T. F. Chan and L. A. Vese, *Active contours without edges*, IEEE Trans. Image Process., 10 (2001), pp. 266–277.

[6] G. Charpiat, O. Faugeras, and R. Keriven, *Approximations of shape metrics and application to shape warping and empirical shape statistics*, Found. Comput. Math., 5 (2005), pp. 1–58.

[7] D. Comaniciu, V. Ramesh, and P. Meer, *Kernel-based object tracking*, IEEE Trans. Pattern Anal. Machine Intell., 25 (2003), pp. 564–577.

[8] D. Cremers, *Dynamical statistical shape priors for level set-based tracking*, IEEE Trans. Pattern Anal. Machine Intell., 28 (2006), pp. 1262–1273.

[9] D. Cremers, *Nonlinear dynamical shape priors for level set segmentations*, in Proceedings of the 2007 IEEE Conference on Computer Vision and Pattern Recognition, Minneapolis, MN, 2007, IEEE Press, Piscataway, NJ, pp. 1–7.

[10] D. Cremers, S. J. Osher, and S. Soatto, *Kernel density estimation and intrinsic alignment for shape priors in level set segmentation*, Int. J. Computer Vision, 69 (2006), pp. 335–351.

[11] D. Cremers, F. R. Schmidt, and F. Barthel, *Shape priors in variational image segmentation: Convexity, Lipschitz continuity and globally optimal solutions*, in Proceedings of the 2008 IEEE Conference on Computer Vision and Pattern Recognition, Anchorage, AK, 2008, IEEE Press, Piscataway, NJ, pp. 1–6.

[12] S. Dambreville, Y. Rathi, and A. Tannenbaum, *Tracking deformable objects with unscented Kalman filtering and geometric active contours*, in Proceedings of the 2006 American Control Conference, Minneapolis, MN, 2006, IEEE Press, Piscataway, NJ, pp. 2856–2861.

[13] B. Friedland, *Control System Design: An Introduction to State-Space Methods*, McGraw–Hill Higher Education, New York, 1985, pp. 259–289.

[14] L. Garcin and L. Younes, *Geodesic matching with free extremities*, J. Math. Imaging Vision, 25 (2006), pp. 329–340.

[15] 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 2001 International Conference on Image Processing, Thessaloniki, Greece, 2001, IEEE Press, Piscataway, NJ, pp. 786–789.
[16] D. Holm, *Geometric Mechanics Part* 2: *Rotating, Translating and Rolling*, Imperial College Press, London, 2008, pp. 109–123.
[17] B. Horn and B. Schunck, *Determining optical flow*, Artificial Intelligence, 17 (1981), pp. 185–203.
[18] J. D. Jackson, A. J. Yezzi, and S. Soatto, *Tracking deformable moving objects under severe occlusions*, in Proceedings of the 43rd IEEE Conference on Decision and Control, Paradise Island, Bahamas, 2004, IEEE Press, Piscataway, NJ, pp. 2990–2995.
[19] M. Kass, A. Witkin, and D. Terzopoulos, *Snakes: Active contour models*, in Proceedings of the Eighth International Conference on Computer Vision, Vancouver, BC, 2001, IEEE Computer Society Press, Piscataway, NJ, pp. 259–268.
[20] E. Klassen, A. Srivastava, W. Mio, and S. H. Joshi, *Analysis of planar shapes using geodesic paths on shape spaces*, IEEE Trans. Pattern Anal. Machine Intell., 26 (2004), pp. 372–383.
[21] J. Malcolm, Y. Rathi, and A. Tannenbaum, *Graph cut segmentation with nonlinear shape priors*, in Proceedings of the International Conference on Image Processing, San Antonio, TX, 2007, IEEE Press, Piscataway, NJ, pp. 365–368.
[22] J. Malcolm, Y. Rathi, and A. Tannenbaum, *Tracking through clutter using graph cuts*, in Proceedings of the 18th British Machine Vision Conference, University of Warwick, UK, 2007, pp. 116–125.
[23] P. Maurel, R. Keriven, and O. Faugeras, *Reconciling landmarks and level sets*, in Proceedings of the 18th IEEE International Conference on Pattern Recognition, Hong Kong, 2006, IEEE Press, Piscataway, NJ, pp. 69–72.
[24] P. S. Maybeck, *Stochastic Models, Estimation and Control*, Vol. I, Academic Press, New York, 1979.
[25] P. W. Michor and D. Mumford, *Riemannian geometries on spaces of plane curves*, J. Eur. Math. Soc., 8 (2004), pp. 1–48.
[26] I. J. Ndiour and P. A. Vela, *Towards a local Kalman filter for visual tracking*, in Proceedings of the 48th IEEE Conference on Decision and Control, Shanghai, 2009, IEEE Press, Piscataway, NJ, pp. 2420–2426.
[27] A. Neumaier and T. Schneider, *Estimation of parameters and eigenmodes of multivariate autoregressive models*, Trans. Math. Software, 27 (2001), pp. 27–57.
[28] M. Niethammer, P. A. Vela, and A. Tannenbaum, *Geometric observers for dynamically evolving curves*, IEEE Trans. Pattern Anal. Machine Intell., 30 (2008), pp. 1093–1108.
[29] N. Papadakis and É. Mémin, *Variational assimilation of fluid motion from image sequence*, SIAM J. Imaging Sci., 1 (2008), pp. 343–363.
[30] E. Pichon, D. Nain, and M. Niethammer, *A Laplace equation approach for shape comparison*, SPIE Medical Imaging, 2 (2006), pp. 24–132.
[31] F. Porikli, O. Tuzel, and P. Meer, *Covariance tracking using model update based on Lie algebra*, in Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, New York, 2006, IEEE Computer Society Press, Piscataway, NJ, pp. 728–735.
[32] 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 the 46th IEEE Conference on Decision and Control, New Orleans, LA, 2007, IEEE Press, Piscataway, NJ, pp. 2681–2688.
[33] Y. Rathi, N. Vaswani, and A. Tannenbaum, *A generic framework for tracking using particle filter with dynamic shape prior*, IEEE Trans. Image Process., 16 (2007), pp. 1370–1382.
[34] M. Rousson and R. Deriche, *A variational framework for active and adaptive segmentation of vector valued images*, in Proceedings of the IEEE Workshop on Motion and Video Computing, Orlando, FL, 2002, IEEE Press, Piscataway, NJ, pp. 56–61.
[35] J. Sethian, *Level Set Methods and Fast Marching Methods*, Cambridge University Press, Cambridge, UK, 1999.
[36] M. Singh, H. Arora, and N. Ahuja, *Robust registration and tracking using kernel density correlation*, in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Washington, DC, 2004, pp. 174–182.
[37] J. Stam, *Real-time fluid dynamics for games*, in Proceedings of the Game Developer Conference, San Jose, CA, 2003.

[38] G. Sundaramoorthi, A. Mennucci, S. Soatto, and A. J. Yezzi, *Tracking deforming objects by filtering and prediction in the space of curves*, in Proceedings of the 48th IEEE Conference on Decision and Control, Shanghai, 2009, IEEE Press, Piscataway, NJ, pp. 2395–2401.

[39] P. H. S. Torr, R. Szeliski, and P. Anandan, *An integrated Bayesian approach to layer extraction from image sequences*, IEEE Trans. Pattern Anal. Machine Intell., 23 (2001), pp. 297–303.

[40] A. Tsai, A. J. Yezzi, W. Wells, C. Tempany, D. Tucker, A. Fan, W. Grimson, and A. Willsky, *A shape-based approach to the segmentation of medical imagery using level sets*, IEEE Trans. Medical Imaging, 22 (2003), pp. 137–154.

[41] N. Vaswani, A. J. Yezzi, Y. Rathi, and A. Tannenbaum, *Time-varying finite dimensional basis for tracking contour deformations*, in Proceedings of the 45th IEEE Conference on Decision and Control, San Diego, CA, 2006, IEEE Press, Piscataway, NJ, pp. 1665–1672.

[42] J. Y. A. Wang and E. H. Adelson, *Layered representation for motion analysis*, in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, New York, 1993, IEEE Press, Piscataway, NJ, pp. 361–366.

[43] J. Xiao and M. Shah, *Motion layer extraction in the presence of occlusion using graph cuts*, IEEE Trans. Pattern Anal. Machine Intell., 27 (2005), pp. 1644–1659.

[44] A. J. Yezzi and A. Mennucci, *Metrics in the Space of Curves*, preprint, arXiv:math.DG/0412454, 2005.

[45] A. J. Yezzi and S. Soatto, *Deformotion: Deforming motion, shape average and the joint registration and approximation of structures in images*, Int. J. Computer Vision, 53 (2003), pp. 153–167.

[46] L. Younes, *Optimal matching between shapes via elastic deformations*, Image Vision Comput. J., 17 (1999), pp. 381–389.