Published in final edited form as:

IEEE Trans Image Process. 2008 December ; 17(12): 2403–2412. doi:10.1109/TIP.2008.2006455.

Segmentation of Tracking Sequences Using Dynamically Updated Adaptive Learning

Oleg Michailovich [Member, IEEE] and

Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, ON N2L 3G1, Canada

Allen Tannenbaum [Member, IEEE]

School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA 30318 USA, and also with the Department of Electrical Engineering, The Technion—Israel Institute of Technology, Haifa, Israel

Oleg Michailovich: olegm@uwaterloo.ca; Allen Tannenbaum: allen.tannenbaum@ece.gatech.edu

Abstract

The problem of segmentation of tracking sequences is of central importance in a multitude of applications. In the current paper, a different approach to the problem is discussed. Specifically, the proposed segmentation algorithm is implemented in conjunction with estimation of the dynamic parameters of moving objects represented by the tracking sequence. While the information on objects' motion allows one to transfer some valuable segmentation priors along the tracking sequence, the segmentation allows substantially reducing the complexity of motion estimation, thereby facilitating the computation. Thus, in the proposed methodology, the processes of segmentation and motion estimation work simultaneously, in a sort of “collaborative” manner. The Bayesian estimation framework is used here to perform the segmentation, while Kalman filtering is used to estimate the motion and to convey useful segmentation information along the image sequence. The proposed method is demonstrated on a number of both computed-simulated and real-life examples, and the obtained results indicate its advantages over some alternative approaches.

Index Terms

Bayesian segmentation; Kalman filtering; motion estimation; tracking

I. Introduction

Image segmentation is the procedure of partitioning the domain of a data image into a number of sub-domains, over which certain statistical properties of the image appear to be homogeneous. Hence, from a certain perspective, image segmentation can be thought of as a way to simplify the image content—the simplification that has proven extremely important and useful in numerous applications including medical imaging [1], [2], surveillance video [3], [4], robotics [5], [6], traffic control [7], [8], and navigation [9], [10].

A multitude of segmentation algorithms have been proposed over the last few decades. Some basic methodologies include histogram-based image segmentation [11], [12]. Using the interpixel dependencies within an image and/or its multiresolution representation is a basic tool used in Bayesian segmentation approaches [13]–[16]. Persistency of the image content in

time and apparent object motion constitute a basis for motion-based segmentation [17]–[19], while specific structural details of an image (e.g., edges) are used in segmentation methods employing active contours [20]–[22]. The correlation with a pre-defined content [23] and color consistency [24] are additional examples of image features used by other segmentation algorithms (for an exhaustive survey on which, the reader is kindly referred to [25]).

Unfortunately, comparing the performance of existing segmentation methods against their computational complexity reveals the well known problem that the accuracy of segmentation and its computational complexity tend to increase *pro rata*. An unfortunate consequence of this fact is that many sophisticated solutions are not applicable in the situations where fast processing is required. In such cases, it may be necessary to bring the performance of relatively simple segmentation methods as close as possible to that of more complex ones. A possible realization of this concept is proposed in the present paper.

The proposed method is based on a sequential Bayesian classification, in which the pixels of a tracking image are ascribed to the segmentation classes whose posterior probability is maximized [26], [27]. Note that the property of such a segmentation scheme of being *sequential* consists in using the posterior probabilities computed for a given tracking image, as prior probabilities for its successor [28], [29]. It should be noted, however, that such a recursion makes sense only if the motion of the tracked objects is relatively slow.

Unfortunately, when the motion is relatively fast, the sequential Bayesian segmentation may no longer be of use because of the errors caused by applying the priors (i.e., the posteriors of preceding images) to incorrect pixels.¹ The above limitation, however, could have been naturally overcome, had the posteriors been properly transformed before being used as prior distributions. It goes without saying that such a transformation, in turn, could be only performed, if the apparent motions related to each of the segmentation classes were known.

Motion estimation in tracking applications is another fundamental problem, which has inspired numerous studies [30], [31]. In the current paper, the apparent motion is represented by means of *optical flows* [32], the estimation of which is nowadays considered as a fundamental problem in numerous applications. A problematic aspect of this estimation, however, is known to be related to its relatively high complexity for the cases when optical flows are discontinuous. Unfortunately, such cases are quite typical in tracking scenarios, in which the motion of a background and those of tracked objects are often relatively simple when taken separately, while the corresponding global motion is discontinuous and, therefore, poorly describable by parametric models of low complexity.

In order to simplify the complexity of motion estimation, we propose to “couple” the latter with the process of image segmentation in the following way. Based on a *predicted* motion field, the posteriors computed for a current tracking image are “warped forward” to form priors for the subsequent image. After the latter has been segmented, the resulting segmentation mask is used to divide (disintegrate) the global optical flow into a number of its simpler components. As a final step, all the “disintegrated” motions are estimated and, subsequently, used to estimate the motion field through *updating* its prediction. It should be noted that the motion estimation above is performed in two stages, *viz.* prediction and update. Hence, the use of a Kalman filter comes naturally for the needs of the proposed approach [33].

It should be noted that the idea of using the prediction property of Kalman filtering in order to transform the past observations of a dynamic scene into its probable future configuration is not new, and it has been exploited in a number of studies [34]–[36]. However, to the best of our

¹Here and hereafter, we occasionally use the terms “priors” and “posteriors” as shorthand substitutes for the prior and posterior probabilities, respectively.

knowledge, using Kalman filtering to integrate the processes of optical-flow-based tracking and Bayesian segmentation (so as to decrease the computational complexity of the former, while improving the performance of the latter) is reported here for the first time.

The paper is organized as follows. Section II briefly reviews the sequential Bayesian classifier used for segmentation, whereas Section III discusses the optical flow model used as well as some methods for its estimation. Kalman filtering is also briefly discussed in this section. Section IV summarizes the overall structure of the algorithm. A number of simulated and real-time examples are demonstrated in Section V. Section VI finalizes the paper with a discussion and conclusions.

II. Segmentation of Tracking Sequences

A. Segmentation via Bayesian Classification

We start with briefly describing the Bayesian segmentation approach of [28] and [29] which constitutes the basis for the segmentation method proposed in the present paper. This approach is based on the assumption that each pixel of the image to be segmented can be ascribed to one of M classes, where M is supposed to be known in advance and fixed. Even though from the practical point of view the latter assumption may seem as a limitation, it is introduced mainly to facilitate the discussion. If necessary, the proposed method can be easily modified to deal with an unknown number of classes, as detailed in [29].

Let $\{C_k\}_{k=1}^M$ denote M segmentation classes, to which the pixels of image $I(\mathbf{n})$ (with $\mathbf{n} \in \Omega \subseteq \mathbb{Z}^2$) need to be ascribed. The segmentation classes C_k , in turn, are characterized by certain *features*, which are usually quantified by real numbers and manipulated as an ordinary Euclidean vector $\theta \in \mathbb{R}^d$. The feature vector is considered to be a random vector, whose (class-) *conditional* probability is different for every C_k , in general. Below, these conditional probabilities are denoted by $p(\theta|C_k)$.

The central notion in the theory of Bayesian classification is that of *posterior probability*. For the case at hand, we are interested in computing the posterior probability of the pixel $\mathbf{n}$ to be in C_k given that the observed feature vector at this pixel is equal to $\theta(\mathbf{n})$. This probability can be computed using the Bayes rule, and it is given by

$$P(z \in C_k | \theta = \theta(\mathbf{n})) = \frac{\Pr_k(\mathbf{n}) p(\theta(\mathbf{n}) | C_k)}{\sum_{k=1}^M \Pr_k(\mathbf{n}) p(\theta(\mathbf{n}) | C_k)} \quad (1)$$

where $\Pr_k(\mathbf{n})$ denotes the prior probability of the pixel $\mathbf{n}$ to be in C_k .

The *Bayesian classifier* ascribes $\mathbf{n}$ to the class, for which the posterior probability (1) is maximal. Namely, having $\theta(\mathbf{n})$ observed, it is decided that $\mathbf{n} \in C_{k^*}$, where

$$k^* = \arg \max_k P(\mathbf{n} \in C_k | \theta = \theta(\mathbf{n})), \quad \forall \mathbf{n}. \quad (2)$$

Note that using more independent attributes of $\{C_k\}_{k=1}^M$ usually improves the performance of Bayesian classification. However, to facilitate the present discussion, we restrict it to the specific case of θ being a *scalar* representing the grey-level values of $I(\mathbf{n})$, viz. $\theta(\mathbf{n}) \equiv I(\mathbf{n})$.

B. Modeling of Conditional Densities

To compute the posteriors in (2), the conditional probabilities $p(\theta|C_k)$ need to be specified first. In the current paper, the probabilities are modeled using the Gaussian Mixture Model (GMM) [37] which is defined as given by

$$p(\theta|C_k) = \sum_{i=1}^{L_k} \pi_i^k \mathcal{N}(\theta|\mu_i^k, \sigma_i^k), \quad \sum_{i=1}^{L_k} \pi_i^k = 1, \quad \forall k \quad (3)$$

where $\mathcal{N}(\theta|\mu_i^k, \sigma_i^k)$ stands for the Gaussian probability density function (*pdf*) with its mean and standard deviation equal to μ_i^k and σ_i^k , respectively. Note that the number of mixture components L_k is class-dependent and it is assumed to be chosen *a priori* for each k so as to optimally balance the accuracy and complexity of the density approximation. By setting $L_k = 1, \forall k$ the setting in [28], [29] is restored.

In real-life scenarios, the hyperparameters $(\pi_i^k, \mu_i^k, \sigma_i^k)_{i=1}^{L_k}$ of each class-conditional pdf can vary in time because of, e.g., changes in the background content and/or changes in illumination conditions. Thus, these hyperparameters should be constantly re-estimated by means of, for instance, the EM algorithm of [38]. We will elaborate on this issue later on. In the meanwhile, it is assumed that these hyperparameters are known and available to be used for computing the conditional densities $p(\theta|C_k)$.

C. Regularization of Posterior Probabilities

In the discrete setting, the data is a sequence of discrete images, which we denote here by $I(\mathbf{n}, t)$ with $\mathbf{n}$ and t being the space and time variables, respectively. The corresponding posteriors can also be arranged as sequences of “images,” which will be denoted below by $P_k(\mathbf{n}, t)$ with $k = 1, 2, \dots, K$. Thus, with this new notation, at time, the Bayesian classifier ascribes the pixel $\mathbf{n}$ to the class C_{k^*} , with $k^* = \arg \max_k P_k(\mathbf{n}, t)$.

Pixelwise application of the classifier can be viewed as an operator that maps the image $I(\mathbf{n}, t)$ to another image (of the same size), which takes its values in the set $\{1, 2, \dots, M\}$ and is commonly referred to as a *segmentation mask*. With slight abuse of notation, we define the segmentation mask $M(\mathbf{n}, t)$ as given by

$$M(\mathbf{n}, t) = \arg \max_k P_k(\mathbf{n}, t) \quad (4)$$

where the maximization is performed *pixelwise*.

Being a pixelwise operation, the segmentation (4) ignores any possible dependencies between the image pixels. Needless to say that this property of the segmentation method can considerably worsen its robustness. Indeed, being a function of measurements, the posteriors (1) inevitably “inherit” some of the properties of the data. For example, if the tracking sequence is noisy, then the posteriors will be noisy as well. As a result, the segmentation mask could appear to be overly fragmented.

To overcome the above deficiency, it was proposed in [28], [29] to “re-introduce” the dependencies between adjacent pixels via prefiltering the posteriors, before the latter are

substituted in (1). Thus, if $\mathbf{S}$ denotes a filtering operator, then the segmentation can be performed according to

$$M(\mathbf{n}, t) = \arg \max_k \mathbf{S}\{P_k(\mathbf{n}, t)\}. \quad (5)$$

The segmentation (5) was demonstrated to provide considerably better results as compared to (4) under a variety of experimental conditions, while remaining relatively fast to compute.

A number of different choices for $\mathbf{S}$ in (5) are possible, which range from ordinary linear filtering to many powerful nonlinear methods [39]–[42]. While linear filtering is certainly the fastest, the others may be preferred, when oversmoothing is undesirable and fine image details (e.g., edges) need to be preserved. As the present study has been aimed at developing a real-time tracking system, an ordinary linear filtering was used throughout the experimental part of the paper. It is to be noted that, in this case, the values of the impulse response of the linear filter should sum to 1 to guarantee that the posteriors are properly normalized.

D. Sequential Bayesian Segmentation

The computation of the posterior probabilities $P_k(\mathbf{n}, t)$ given by (1) requires knowing the corresponding prior probabilities $\Pr_k(\mathbf{n}, t)$. Typically, the priors are elicited from either training data or some antecedent knowledge on morphological details of the object being tracked. In dynamic scenarios, however, it seems to be very natural and advantageous for computational reasons to use, so called, *sequential* priors, which are nothing else but time-delayed posteriors. Formally

$$\Pr_k(\mathbf{n}, t) = P_k(\mathbf{n}, t - 1), \quad \forall k \quad (6)$$

with $\Pr_k(\mathbf{n}, 0) = M^{-1}$.

Using the sequential priors (6) allows fusing useful segmentation information along the time axis. However, these priors makes sense only if the motion of tracked objects is negligible over the time span of two successive images. Otherwise, using the priors updated according to (6) can result in erroneous segmentation.

To overcome the above deficiency, we propose a different rule for dynamically updating the prior probabilities. Formally, this rule can be expressed as given by

$$\Pr_k(\mathbf{n}, t) = P_k(\mathbf{T}_t[\mathbf{n}], t - 1), \quad \forall k \quad (7)$$

where $\mathbf{T}_t[\cdot]$ is a *map* relating $I(\mathbf{n}, t - 1)$ to its follower $I(\mathbf{n}, t)$ according to $I(\mathbf{T}_t[\mathbf{n}], t - 1) \simeq I(\mathbf{n}, t)$. For the convenience of referencing, the update rule of (7) will be referred below to as *dynamically updated adaptive learning* (DUAL).

It should be noted that, in the current paper, the map $\mathbf{T}_t[\cdot]$ is defined by an optical flow, that implies the existence of $u(\mathbf{n}, t)$ such that $\mathbf{T}_t[\mathbf{n}]: \mathbf{n} \mapsto \mathbf{n} - u(\mathbf{n}, t)$. However, as it was pointed out in the introductory sections of the paper, in many cases of practical interest, the order of a global map should be kept sufficiently high to properly account for the discontinuities of the composite motion. As a result, estimation of the global flow $u(\mathbf{n}, t)$ could be computationally too intensive to be realizable in a real-life scenario.

The problem above, however, could be easily overcome, if segmentation masks for the images $I(\mathbf{n}, t-1)$ and $I(\mathbf{n}, t)$ were available. The segmentation breaks down the domain of image definition Ω into M disjoint subsets $\{\Omega_k\}_{k=1}^M$, so that each of the subsets supports a distinct segmentation class, which is presumed to be associated with a *local* optical flow describable by a low order model. In this case, one can define $\{\mathbf{T}_t^k\}_{k=1}^M$ to be a set of maps and $\{u_k(\mathbf{n}, t)\}_{k=1}^M$ to be a set of associated optical flows, such that each $\mathbf{T}_t^k$ (or, equivalently $u_k(\mathbf{n}, t)$) “acts” on a distinct subset of Ω

$$\mathbf{T}_t^k[\mathbf{n}]: \mathbf{n} \mapsto \mathbf{n} - u_k(\mathbf{n}, t), \quad \forall k. \quad (8)$$

To compute the prior probabilities $\Pr_k(\mathbf{n}, t)$ using DUAL requires knowing the local maps $\{\mathbf{T}_t^k\}_{k=1}^M$, which are generally unknown and, hence, should be estimated from the data. As it was pointed out earlier, such an estimation procedure would be possible, if the images $I(\mathbf{n}, t-1)$ and $I(\mathbf{n}, t)$ were given along with their corresponding segmentation masks $M(\mathbf{n}, t-1)$ and $M(\mathbf{n}, t)$. The latter, however, could be computed only if the maps $\{\mathbf{T}_t^k\}_{k=1}^M$ were known. In order to break this logical circle, we replace the local maps by their *predictions* derived from the past observations of the dynamic system. The prediction is performed by a Kalman filter, which constitutes the core of the motion estimation method we describe next.

III. Motion Estimation by Kalman Filtering

A. Affine Optical Flow

The problem we address next is that of estimating the apparent motion of tracked objects given a sequence of images $\{I(\mathbf{n}, t)\}_{t \geq 0}$. The estimation is based on assuming any pair of successive images to be related according to $I(\mathbf{n} - u_t(\mathbf{n}, t), t-1) \simeq I(\mathbf{n}, t)$. This implies that any subsequent image can be represented as a locally translated version of its predecessor, where the local translations are governed by the optical flow $u(\mathbf{n}, t)$, which is a function of time t . In order to facilitate the discussion, the optical flow here is temporarily assumed to be global.

The problem of estimating the optical flow $u(\mathbf{n}, t)$ is ill-posed, in general. Therefore, in order to render its solution stable and unique, all the admissible optical flows should be restricted to some “target” class via imposing appropriate smoothness constraints on their spatial and/or temporal behavior. An alternative way to regularize the solution is via parametrization. Such parametric models have proven advantageous in practice since they allow reducing the order of the motion model, thereby facilitating the computations. More specifically, one can restrict the solution to belong to a family parameterized by a vector of real parameters $a_t \in \mathbb{R}^q$. In this case, the datum of the sequence $\{a_t\}_{t \geq 0}$ is equivalent to the datum of the optical flow itself. As the current work is aimed at developing a system for real-time tracking, the parametrization seems to be a reasonable way to regularize estimating the optical flow.

When the tracking of rigid targets is under consideration, assuming the displacement field $u(\mathbf{n}, t)$ to be affine may be quite reasonable, in which case $a_t \in \mathbb{R}^6$ and

$$u(\mathbf{n}, t) = u(\mathbf{n}|a_t) = \mathbf{Z}a_t, \quad \mathbf{Z} = \begin{pmatrix} n_1 & n_2 & 0 & 0 & 1 & 0 \\ 0 & 0 & n_1 & n_2 & 0 & 1 \end{pmatrix} \quad (9)$$

with n_1 and n_2 being the first and the second coordinate of $\mathbf{n}$, viz. $\mathbf{n} = (n_1, n_2)$. A typical example here is the tracking of military targets (e.g., tanks, armored troop-carriers, aircrafts) using an airborne SAR or CCD camera freely moving in 3-D.² Then, assuming the amplitude of the displacement field $u(\mathbf{n}, t)$ to be relatively small, its *least square* (LS) estimate can be shown to be given by [43], [44]

$$a_t^* = R_t^{-1} P_t \quad (10)$$

where $R_t = \sum_{\mathbf{n} \in \Omega} A(\mathbf{n}, t) A(\mathbf{n}, t)^T$, $P_t = \sum_{\mathbf{n} \in \Omega} A(\mathbf{n}, t) \partial^t I(\mathbf{n}, t)$, and $A(\mathbf{n}, t) = \nabla I(\mathbf{n}, t) \mathbf{Z}$ with $\nabla I(\mathbf{n}, t)$ and $\partial^t I(\mathbf{n}, t)$ standing for the spacial and temporal derivatives of $I(\mathbf{n}, t)$, respectively.

Finally, we note that the estimate (10) is based on assuming the optical flow to have relatively small magnitudes. However, when rapidly moved objects are tracked, the magnitude of the optical flow can be large even for relatively high image-acquisition rates. Unfortunately, when this happens, the solution by (10) may no longer provide a useful result. In this case, using the multiscale approach of [45] seems to be an effective alternative way to compute a_t^* on account of only marginal increase in the amount of required computations.

B. Kalman Filtering of Affine Parameters

Equation (10) provides a solution to the problem of estimating affine optical flow between any pair of successive images within the sequence $\{I(\mathbf{n}, t)\}_{t \geq 0}$. However, if one needs to predict the affine parameters a_t given the input images up to the time $t - 1$ inclusive, this equation by itself is of little use, since it does not take into account the temporal coherence of the motion field. Thus, in order to predict the optical flow, the temporal “tendencies” within the sequence $\{a_t\}_{t \geq 0}$ should be duly brought into consideration.

To account for the temporal coherence of the affine parameters, we describe their evolution using the Kalman system given by

$$\begin{cases} a_t = a_{t-1} + e_t \\ a_t^* = a_t + \xi_t \end{cases} \quad (11)$$

where e_t and ξ_t stand for the state and measurement noises, respectively. These noises are assumed to be mutually independent, zero-mean, Gaussian noises having time-independent covariances $Q_t = \mathcal{E}\{e_t e_t^T\}$ and $W_t = \mathcal{E}\{\xi_t \xi_t^T\}$. It is worthwhile noting that the system (11) suggests that the affine parameters a_t pertaining to time t are equal to their time-delayed version a_{t-1} contaminated by the state noise e_t , while the estimated parameters a_t^* are considered to be a noisy measurement of the true parameters a_t .

The system (11) can be efficiently solved via Kalman filtering [33]. Although the theory of Kalman filtering is very well known, there is a particular reason to recall the filter structure, since it plays a crucial role in the proposed tracking algorithm. Specifically, the Kalman filter estimates the state a_t in two basic steps. First, a prediction $\tilde{a}_t$ of a_t is computed. For the case at hand, the prediction equation is given by

²Note that such an assumption requires perspective effects to be relatively negligible.

$$\bar{a}_t = a_{t-1}, \quad \bar{\Sigma}_t = \Sigma_{t-1} + Q_t \quad (12)$$

where Σ_{t-1} denotes the posterior error covariance computed at time $t-1$ and $\bar{\Sigma}_t$ is its prediction at time t . As the second step, the prediction is updated using the information brought in with a new observation obtained by (10)

$$a_t = \bar{a}_t + \frac{\bar{\Sigma}_t}{\bar{\Sigma}_t + W_t} (a_t^* - \bar{a}_t), \quad \Sigma_t = \left(I - \frac{\bar{\Sigma}_t}{\bar{\Sigma}_t + W_t} \right) \bar{\Sigma}_t. \quad (13)$$

The fact that the prediction appears explicitly as an intrinsic part of the Kalman filter forms the main property underlying the dynamic segmentation method proposed in this paper. In particular, the predicted parameters of the optical flow (12) are used by DUAL to propagate the posteriors and to perform the segmentation, while the resulting segmentation masks are used to decompose the global motion field into its affine components, followed by estimating the parameters of the latter. Having the affine parameters estimated, the Kalman filter regards them as new observations of the true states and performs the update according to (13).

C. Estimation of Individual Motion Components

Before summarizing the overall structure of the proposed algorithm, a few words need to be said regarding the method we use to “isolate” the local (affine) components of the global motion for the purpose of their estimation. Note that the basic assumption used here is that each local motion can be associated with a distinct segmentation class.

Let the total number of objects (including background) be equal to M , and the support of the k^{th} object in two successive images $I(\mathbf{n}, t)$ and $I(\mathbf{n}, t+1)$ be $\Omega_t^k \subseteq \Omega$ and $\Omega_{t+1}^k \subseteq \Omega$, correspondingly. Then, the k^{th} affine model can be estimated by applying the procedure specified in the preceding section to the *weighted* images $I(\mathbf{n}, t)W_k(\mathbf{n}, t)$ and $I(\mathbf{n}, t+1)W_k(\mathbf{n}, t+1)$ (instead of $I(\mathbf{n}, t)$ and $I(\mathbf{n}, t+1)$, respectively), where the weighting functions are computed according to

$$W_k(\mathbf{n}, t) = \begin{cases} 1, & \text{if } \mathbf{n} \in \Omega_t^k \\ 0, & \text{if } \mathbf{n} \notin \Omega_t^k \end{cases} \quad W_k(\mathbf{n}, t+1) = \begin{cases} 1, & \text{if } \mathbf{n} \in \Omega_{t+1}^k \\ 0, & \text{if } \mathbf{n} \notin \Omega_{t+1}^k \end{cases}. \quad (14)$$

It should be noted that the weighting functions given by (14) are computed based on the related segmentation masks which are estimated quantities by themselves. This implies that, due to estimation errors, the supports of the weighting functions cannot be ideally aligned with the respective regions of local motion. Moreover, the weighting functions are discontinuous—the property that can also give rise to numerical inaccuracies.

Fortunately, the above deficiencies can be substantially alleviated via slightly extending the support of $W(\mathbf{n}, t)$ and $W(\mathbf{n}, t+1)$ by means of morphological dilation [46], followed by linear filtering the results thus obtained. In the present paper, the dilation has been performed using a disk-shaped structural element with the radius of 4 to 6 pixels.

IV. Dual: Overall Structure

The proposed segmentation algorithm is executed recursively. At each step, the segmentation mask pertaining to $I(\mathbf{n}, t)$ and the parameters of affine optical flows relating $I(\mathbf{n}, t)$ with its predecessor $I(\mathbf{n}, t-1)$ are supposed to have been already estimated. At the output, the same information is recovered for $I(\mathbf{n}, t+1)$. Below, we specify a sequence of procedures by which the segmentation using DUAL can be implemented.

- Step 1: [*The first stage of the Kalman filtering*] The set $\{a_t^k\}_{k=1}^M$ of the affine parameters corresponding to time t is used to predict the affine parameters $\{a_{t+1}^k\}_{k=1}^M$ using (12). This step results in the set of predicted parameters $\{\bar{a}_{t+1}^k\}_{k=1}^M$.
- Step 2: The predicted parameters $\{\bar{a}_{t+1}^k\}_{k=1}^M$ are used to compute the related optical flows $\{u(\mathbf{n}|\bar{a}_{t+1}^k)\}_{k=1}^M$. Subsequently, the posteriors $\{P_k(\mathbf{n}, t)\}_{k=1}^M$ are pushed forward by the predicted optical flows according to the methodology of DUAL. This step results in the set of priors $\{\text{Pr}_k(\mathbf{n}, t+1)\}_{k=1}^M$ for the image $I(\mathbf{n}, t+1)$.
- Step 3: [*The Bayesian segmentation*] The posterior probabilities of the segmentation classes in $I(\mathbf{n}, t+1)$ are computed according to (1), and, subsequently, used to estimate the segmentation mask $M(\mathbf{n}, t+1)$ using (5).
- Step 4: The weighting functions $\{W_k(\mathbf{n}, t+1)\}_{k=1}^M$ are computed based on $M(\mathbf{n}, t+1)$ using (14) (followed by morphological dilation and smoothing, if necessary). Subsequently, the images $I(\mathbf{n}, t)$ and $I(\mathbf{n}, t+1)$ are multiplied by the elements of $\{W_k(\mathbf{n}, t)\}_{k=1}^M$ and $\{W_k(\mathbf{n}, t+1)\}_{k=1}^M$, respectively, and the resulting weighted images are used to estimate the local optical flows using (10). This step results in the set of estimated affine parameters $\{a_{t+1}^{*k}\}_{k=1}^M$.
- Step 5: [*The second stage of the Kalman filtering*] The predicted $\{\bar{a}_{t+1}^k\}_{k=1}^M$ and the “measured” $\{a_{t+1}^{*k}\}_{k=1}^M$ parameters are used to update the estimation of the affine parameters according to (13). This step results in the set of updated parameters $\{a_{t+1}^k\}_{k=1}^M$, which define the optical flow “connecting” the images $I(\mathbf{n}, t)$ and $I(\mathbf{n}, t+1)$.
- Step 6: [*Optional “backing” stage*] The steps 2 and 3 are repeated for the updated parameters $\{a_{t+1}^k\}_{k=1}^M$ instead of the predicted parameters $\{\bar{a}_{t+1}^k\}_{k=1}^M$.
- Step 7: The segmentation mask $M(\mathbf{n}, t+1)$ is used to classify the pixels of $I(\mathbf{n}, t+1)$ into M groups (as per the number of classes), which are used to re-estimate the hyperparameters $\{(\pi_i^k, \mu_i^k, \sigma_i^{Lk})_{i=1}^M\}_{k=1}^M$ using the EM algorithm [38].

Finally, we note that the computational complexity of one iteration of DUAL is mainly defined by Steps 2, 3, 4, and 7. Specifically, pushing forward the posteriors $\{P_k(\mathbf{n}, t)\}_{k=1}^M$ requires applying an interpolation procedure. In practice, accurate and stable results have been obtained using linear interpolation, which can be performed with linear complexity in the number of image samples. The Bayesian classification of Step 3 can be carried out with linear complexity as well, if the class-conditional densities are computed by means of lookup tables. Estimating the affine parameters for each of the M segmentation classes requires computing the partial derivatives of the weighted images. The number of such computations depends on M as well

as on the number of multiscale levels involved in computation of $\{a_{t+1}^{*k}\}_{k=1}^M$ using the pyramid-based approach of [45]. However, since the LS estimation of the affine parameters requires inverting only 6×6 matrices, the complexity of Step 4 still remains linear in the number of image samples. Finally, the complexity of Step 7 is mainly defined by $\{L_k\}_{k=1}^L$ as well as by the number of EM iterations. In practice, however, the latter has rarely exceeded five iterations to update the hyperparameters with more than 1% accuracy. Note that the initial values of the hyperparameters are assumed here to be provided in the beginning of the DUAL recursion. These values could be, for instance, learned based on some previous observations of the scene of interest or estimated based on the results of either manual or semi-automatic segmentation.

V. Experimental Results

A. Computer-Simulated Experiments

Computer-simulated experiments are usually the first validation stage intended to assess the performance of a given algorithm under controllable conditions. To this end, a number of tracking sequences were synthesized using two “aircraft” objects and a background of the form shown on the left and right subplots of Fig. 1, respectively. The amplitude of the background was set to be such that its largest value comes to 95% of the objects’ grey level (with the latter being normalized to unity). It has been done so as to make the “aircrafts” poorly recognizable over the bright spots of the background, thereby challenging the segmentation.

The motions of the “aircrafts” were simulated by applying to their images two sequences of affine optical flows, whose parameters were evolved in accordance with (11). The initial state a_0 for the first object was set to be equal to $[0.134, -0.5, 0.5, 0.134, 0, 0]^T$, which corresponds to a rotation through the angle of 30° . For the second object, a_0 corresponded to a rotation through the angle of -30° . For both objects, the state noise covariance Q_t was set to be a diagonal matrix with its diagonal equal to $[1 \ 1 \ 1 \ 1 \ 0.1 \ 0.1] \cdot 10^{-3}$, $\forall t$. The measurement noise covariance W_t was set to be an identity matrix proportional to $5 \cdot 10^{-3}$, $\forall t$.

The resulting images of the “aircrafts” were superimposed on the background, at the same time when the latter was subjected to random shifts intended to mimic the camera egomotion. The coordinates of the shifts were defined to be i.i.d. Gaussian random variables with zero mean and variance equal to 2 pixels. The resulting images were contaminated by white Gaussian noise giving rise to three different levels of SNR, viz. 10, 15, and 20 dB. The image size was set to be 256×256 , and each simulated sequence consisted of 25 images.

As a first part of the experiment, the simulated sequences were subjected to the Bayesian segmentation procedure of [28], [29] using the sequential priors (6). For convenience, this algorithm is referred below as “static Bayesian.” The conditional densities of the object and background classes were defined according to (3) with the number of mixture components being 1 for the “aircrafts” and 2 for the background. The smoothing operator $\mathbf{S}$ in (5) was defined to be linear filtering using a fourth-order (separable) binomial filter. As the next step, the tracking sequences were processed by the Bayesian classifier (5) using the DUAL rule (7) (below, we refer to this method as “dynamic Bayesian”). Here, the segmentation parameters were set identically to the previous case. Moreover, at each iteration, the weighting functions (14) have been morphologically dilated using a disk-shaped structural element with a radius of 4 pixels, followed by smoothing the resulted functions by convolving the latter with a fourth-order binomial filter. In addition, estimation of the dynamic parameters was initialized with zero initial conditions and the error covariance equal to $\Sigma_0 = 10^{-2} \mathbf{I}_{6 \times 6}$ for both objects, where $\mathbf{I}_{6 \times 6}$ is a 6×6 identity matrix.

A typical result for the case of $\text{SNR} = 20$ dB is shown in Fig. 2, where the upper row of subplots depicts a subset of the simulated images corresponding to $t = [3, 6, 9, 12, 15, 18, 21, 24]$, while the middle and the lower rows show the segmentation masks obtained with “static Bayesian” and “dynamic Bayesian,” respectively. One can see that “static Bayesian” provides unacceptable results, since none of its segmentation masks correctly represents the shape of the object being tracked. On the other hand, the segmentation masks computed by “dynamic Bayesian” correctly represent the content of the simulated images as well as the shapes of both “aircrafts” for all t .

Since the simulated images of the “aircrafts” are binary, the segmentation masks obtained in this manner can be used for quantitatively comparing the simulation results. Particularly, if the “aircrafts” and the background classes are labeled by 1 and 0, respectively, the resulting segmentation masks can be regarded as estimates of the images of the “aircrafts”. In this case, the mean-squared error (MSE) criterion can be used to evaluate the algorithm’s performance. A typical behavior of the MSE as a function of time is shown in Fig. 3 for $\text{SNR} = 20$ dB. One can see that “dynamic Bayesian” (dotted-solid line) provides considerably more accurate estimation results than “static Bayesian” (solid line). Moreover, while the MSE of “static Bayesian” increases as a function of time, the MSE of “dynamic Bayesian” slightly decreases due to the convergence of the Kalman filter.

Finally, Subplots A1–A5 of Fig. 4 show the segmentation masks (corresponding to the same t as above) computed using “dynamic Bayesian” for $\text{SNR} = 15$ dB, while Subplots B1–B5 of the same figure show the masks for the case $\text{SNR} = 10$ dB. One can see that, while the shape of the “larger aircraft” is recovered with relatively high accuracy independently of the noise level, the reconstruction of the “smaller aircraft” degrades, as the SNR decreases. This happens for two main reasons. First, the smaller the object, the smaller the numerical support for estimating its affine parameters and, hence, the higher the error of estimating the related optical flow. Second, the smaller the object is, the higher is the probability of its being misaligned with the corresponding (dynamically updated) priors due to the errors in estimating the affine parameters. Despite the above limitations, the proposed algorithm is still capable of reliably tracking relatively small targets as it is further demonstrated in the section that follows.

B. Experiments With Real Data

As a next validation step, the algorithm performance was tested using real-life video sequences. The sequences were acquired using an optical (digital) camera mounted on a training aircraft. During the data acquisition, the latter was instructed to follow another, freely-maneuvering aircraft so as to capture its images over dynamically changing background. Note that this experiment was conducted as part of the project for developing a system for automated control of unmanned air vehicles (UAV) during a flight in formation. In such applications, the information provided by a segmentation procedure is vital for defining a number of important parameters of the formation structure (e.g., relative positions of the UAVs and their mutual distances). Moreover, in such applications, occlusions are rare, while the main difficulties are usually related to the existence of clutter noise as well as to the ego-motion of tracking cameras. Note that the camera ego-motion (being caused by aircraft’s vibrations and wobbles) is transformed into the motion of background which makes the spatial occurrence of clutter noise be arbitrary, thereby further worsening the conditions under which the segmentation is to be performed.

Fig. 5 shows a subset of typical tracking images. The image acquisition rate was set to be approximately equal to 30 fps, and the figure shows the frames number 15, 30, 45, 60, 75, 90, 105, 120, 135, 150, 165, 180 (if counted in a normal left-to-right, up-to-down direction). One can see that, in virtually all the images, the target (i.e., the followed aircraft) is hardly recognizable over its background, being represented by the gray-levels that are very close to

those representing the underlying terrain. Consequently, due to the relatively strong ego-motion of the camera and a fast motion of the followed aircraft, the priors elicited using the sequential rule (6) have been of no use and, as a result, the “static Bayesian” approach failed to provide a stable segmentation of the tracking sequences.³ On the contrary, “dynamic Bayesian” provided stable and useful segmentations of all data sequences. To perform the segmentation, the object and background classes were modeled using the Gaussian mixture model (3) with one and three mixture components, respectively. The hyperparameters of the conditional densities were updated dynamically according to Stage 7 of Section IV. At each iteration, the weighting functions (14) have been morphologically dilated using a disk-shaped structural element having a radius of 5 points, followed by smoothing the resulted functions by linearly filtering the latter with a fourth-order (separable) binomial filter. As to the motion estimation part, the covariance matrices were set to be $Q_t = W_t = 0.2 \cdot \mathbf{I}_{6 \times 6}$, $\forall t$. The posterior error covariance was set to be equal to $\Sigma_0 = 0.01 \cdot \mathbf{I}_{6 \times 6}$. Fig. 6 shows the results of segmentation of the tracking images of Fig. 5 (where the magenta contour indicates the segmentation boundary; for a color version of the figure visit <http://www.iee-explore.ieee.org/>). Comparing these figures, one can see that the segmentation provided by “dynamic Bayesian” well agrees with the actual support of the tracked aircraft.

Finally, the same tracking sequences were processed using the motion-based segmentation method of [17], where it is used for deriving a layered representation of video sequences. This method proceeds in two main stages, during the first of which a *dense* optical flow field corresponding to a given frame is computed. Subsequently, the field is divided into small blocks whose related (local) affine models are estimated and, subsequently, clustered by means of the k-means algorithm. The latter yields a number of “motion centroids” which are used for classification of the vectors of the dense optical flow. In the present study, the dense optical flow was estimated based on 3×3 pixel neighborhoods, while the size of the local blocks was set to be equal to 5×5 . Moreover, 50% overlap between the blocks were allowed to increase the number of affine vectors related to the motion of the tracked aircraft. The number of motion clusters was set to be equal to 2. The remaining parameters of the algorithm were set according to the guidelines provided in the original reference (see [17, Section V] for more details). The resulting segmentation is shown in Fig. 7, which reveals an unfortunate drawback of the reference method that consists in its being incapable of reliably tracking small objects/targets. The difference in performances of the proposed and reference methods can be further seen in Fig. 8, which shows the original frame #150 (left sub-plot) along with its segmentation by the motion-based approach of [17] (middle subplot) and by “dynamic Bayesian” (right sub-plot).

VI. Discussion and Conclusions

The present study introduced a computationally efficient method for segmenting the sequences of tracking images. The segmentation was performed within the Bayesian estimation framework, which is known to be centered around the concept of prior probabilities, and where has always been a concern about the way the latter are elicited. In fact, the present study advocates the point that, in dynamic scenarios, some valuable priors can come naturally from previous observations of the scene of interest.

The central part of the proposed method is the DUAL rule used for priors elicitation. This rule can be viewed as an extension of the classical method of sequentially learning the prior probabilities [28] and [29]. Specifically, while the standard approach substitutes the posteriors of the preceding images for the priors of their followers, the DUAL rule transforms the posteriors according to the apparent motion before the substitution is done. This allows the

³In fact, we did not succeed to find a set of parameters, which could render “static Bayesian” stable for even few successive frames.

Bayesian segmentation using DUAL to perform considerably better as compared to the “dynamics-in-different” approaches of [28], [29].

To estimate the motion of tracked objects, linear Kalman filtering was used. It should be noted, however, that assuming the state and measurement models to be linear and the noise processes in (11) to be white Gaussian may be restrictive in general. Although the adequacy of our assumptions has been verified experimentally, the linear Kalman filter can be potentially replaced by particle filtering [48], which is capable of estimating the motion parameters for (reasonably) arbitrary types of models and noises. Moreover, the choice of a model for optical flow is application dependent as well, and, hence, the affine model used in this study may be replaced by a different model, which is more appropriate for an application at hand.

In the case when more than one target is tracked, the problem of association of different segments within two successive images becomes particularly important. In the current paper, the image segments are associated based on the concept of a minimal distance, while the latter is defined as a sum of two components, *viz.* “geometric” and “probabilistic.” For each pair of the image segments, the first component is computed as either the Hausdorff distance between the segments or the Euclidean distance between their centroids. At the same time, the “probabilistic” component is computed as a distance between the empirical pdf’s of the feature vector corresponding to the segments. Such a distance can be defined, for example, based on either the Kullback-Leibler divergence or the Bhattacharyya distance. Subsequently, segment k_1 in the current image is associated with segment k_2 in the subsequent image if their distance is minimal among all possible candidates.

In the current study, the Bayesian segmentation algorithm discriminated between tracked objects and background based on the information contained in pixel values. It was shown that, in this form, the algorithm is capable of reliably tracking fast-maneuvering targets contaminated by clutter noise. However, if the proposed method is to be used for different applications, it should be thoroughly investigated which features should be passed to the Bayesian classifier to make the segmentation less sensitive to a wider class of disturbances.

Acknowledgments

This work was supported in part by grants from the National Science Foundation, the Air Force Office of Sponsored Research, the Army Research Office, MURI, MRI-HEL, and a Discovery grant from NSERC—the Natural Sciences and Engineering Research Council of Canada. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Anil Kokaram.

References

1. Tsai A, Yezzi A, Wells W, Tempany C, Tucker D, Fan A, Grimson WE, Willsky A. A shape-based approach to the segmentation of medical imagery using level sets. *IEEE Trans Med Imag* Feb;2003 22(2):137–154.
2. Grau V, Mewes A, Alcaniz M, Kikinis R, Warfield SK. Improved watershed transform for medical image segmentation using prior information. *IEEE Trans Med Imag* Apr;2004 23(4):447–458.
3. Stringa E, Regazzoni CS. Real-time video-shot detection for scene surveillance applications. *IEEE Trans Image Process* Jan;2000 9(1):69–79. [PubMed: 18255373]
4. Haritaoglu I, Harwood D, Davis LS. W⁴: Real-time surveillance of people and their activities. *IEEE Trans Pattern Anal Mach Intell* Aug;2000 22(8):809–830.
5. Rimey RD, Cohen FS. A maximum-likelihood approach to segmenting range data. *IEEE Trans Robot Autom* Jun;1988 4(3):277–286.
6. Nair D, Aggarwal JK. Moving obstacle detection from a navigating robot. *IEEE Trans Robot Autom* Jun;1998 14(3):404–416.

7. Kato J, Watanabe T, Joga S, Rittscher J, Blake A. An HMM-based segmentation method for traffic monitoring movies. *IEEE Trans Pattern Anal Mach Intell* Sep;2002 24(9):1291–1296.
8. Mellia M, Meo M, Casetti C. Tcp smart framing: A segmentation algorithm to reduce TCP latency. *IEEE/ACM Trans Netw* Apr;2005 13(2):316–329.
9. Bhanu B, Holben R. Model-based segmentation of FLIR images. *IEEE Trans Aerosp Electron Syst* Jan;1990 26(1):2–11.
10. Reed S, Petillot Y, Bell J. An automatic approach to the detection and extraction of mine features in sidescan sonar. *IEEE J Ocean Eng* Jan;2003 28(1):90–105.
11. Glasbey CA. An analysis of histogram-based thresholding algorithms. *Graphical Models and Image Process* 1993;55:532–537.
12. Sahoo PK, Soltani S, Wang AKC. A survey of thresholding techniques. *Comput Vis, Graph, Image Process* 1988;41:233–260.
13. Mumford D, Shah J. Optimal approximations by piecewise smooth functions and associated variational problems. *Commun Pure Appl Math* 1989;42(5):577–685.
14. Chang MM, Tekalp AM, Sezan MI. Simultaneous motion estimation and segmentation. *IEEE Trans Image Process* Sep;1997 6(9):1326–1333. [PubMed: 18283022]
15. Unser M. Texture classification and segmentation using wavelet frames. *IEEE Trans Image Process* Apr;1995 11(4):1549–1560. [PubMed: 18291987]
16. Aujol JF, Aubert G, Blanc-Feraud L. Wavelet-based level set evolution for classification of textured images. *IEEE Trans Image Process* Dec;2003 12(12):1634–1641. [PubMed: 18244717]
17. Wang JYA, Adelson EH. Representing moving images with layers. *IEEE Trans Image Process* Sep;1994 3(9):625–638. [PubMed: 18291956]
18. Bouthemy P, Francois E. Motion segmentation and qualitative dynamic scene analysis from an image sequence. *Int J Comput Vis* 1993;10:157–182.
19. Diehl N. Object-oriented motion estimation and segmentation in image sequences. *Signal Process: Image Commun* 1991;3:23–56.
20. Chan TF, Vese LA. Active contours without edges. *IEEE Trans Image Process* Feb;2001 10(2):266–277. [PubMed: 18249617]
21. Yezzi A, Kichenassamy S, Kumar A, Olver P, Tannenbaum A. A geometric snake model for segmentation of medical imagery. *IEEE Trans Med Imag* Apr;1997 16(2):199–209.
22. Siddiqi K, Lauziere YB, Tannenbaum A, Zucker SW. Area and length minimizing flows for shape segmentation. *IEEE Trans Image Process* Mar;1998 7(3):433–443. [PubMed: 18276263]
23. Xie J, Jiang Y, Tsui HT. Segmentation of kidney from ultrasound images based on texture and shape priors. *IEEE Trans Med Imag* Jan;2005 24(1):45–57.
24. Sharbek, W.; Koschan, A. Color Segmentation Survey. Univ. Berlin; Berlin, Germany: 1994. Tech. Rep
25. Pal NR, Pal SK. A review on image segmentation techniques. *Pattern Recognit* 1994;26(9):1277–1294.
26. Berger, JO. Statistical Decision Theory and Bayesian analysis. New York: Springer-Verlag; 1985.
27. Carlin, BP.; Louis, TA. Monographs on Statistics and Applied Probability. Vol. 69. New York: Chapman & Hall; 1996. Bayes and empirical Bayes methods for data analysis.
28. Teo P, Sapiro G, Wandell B. Creating connected representations of cortical gray matter for functional MRI visualization. *IEEE Trans Med Imag* Dec;1997 16(6):852–863.
29. Haker S, Sapiro G, Tannenbaum A. Knowledge-based segmentation of SAR data with learned priors. *IEEE Trans Image Process* Feb;2000 9(2):299–301. [PubMed: 18255401]
30. Furht, B.; Greenberg, J.; Westwater, R. Motion Estimation Techniques for Video Compression. Norwell, MA: Kluwer; 1996.
31. Bar-Shalom, Y.; Li, XR. Estimation and Tracking: Principles, Techniques, and Software. Boston, MA: Artech House; 1993.
32. Horn BKP, Schunck BG. Determining optical flow. *Artif Intell* Aug;1981 17:185–203.
33. Zarchan P, Musoff H. Fundamentals of Kalman filtering: A practical approach. *Progr Astronaut Aeronaut* 2000;190

34. Petillot Y, Ruiz IT, Lane DM. Underwater vehicle obstacle avoidance and path planning using a multi-beam forward looking sonar. *IEEE J Ocean Eng* Apr;2001 26(2):240–251.
35. Huang, Y.; Huang, TS.; Niemann, H. Segmentation-based object tracking using image warping and Kalman filtering. presented at the IEEE Int. Conf. Image Processing; 2002.
36. Unal G, Krim H, Yezzi A. Fast incorporation of optical flow into active polygons. *IEEE Trans Image Process* Jun;2005 14(6):745–759. [PubMed: 15971774]
37. Everitt, BS.; Hand, DJ. *Finite Mixture Distributions*. New York: Chapman & Hall; 1981.
38. Xu L, Jordan MI. On convergence properties of the EM algorithm for Gaussian mixtures. *Neural Comput* Jan;1996 8(1):129–151.
39. Gallagher NC, Wise GL. A theoretical analysis of the properties of the median filter. *IEEE Trans Acoust, Speech, Signal Process* Jun;1981 ASSP-29(6):1136–1141.
40. Donoho DL. De-noising by soft-thresholding. *IEEE Trans Inf Theory* May;1995 41(3):613–627.
41. Rudin L, Osher S, Fatemi E. Nonlinear total variation based noise removal algorithms. *Phys D* 1992;60:259–268.
42. Sapiro G, Tannenbaum A. On invariant curve evolution and image analysis. *Indiana Univ J Math* 1993;42:985–1009.
43. Robinson D, Milanfar P. Fast local and global projection-based methods for affine motion estimation. *J Math Imag Vis* 2003;18:35–54.
44. Periaswamy S, Farid H. Elastic registration in the presence of intensity variations. *IEEE Trans Med Imag* Jul;2003 22(7):865–874.
45. Berger JR, Anandan P, Hanna KJ, Hingorani R. Hierarchical model-based motion estimation. *Proc ECCV* 1992:237–252.
46. Giardina, CR.; Dougherty, ER. *Morphological Methods in Image and Signal Processing*. Englewood Cliffs, NJ: Prentice-Hall; 1998.
47. Michailovich O, Tannenbaum A. Dynamic de-noising of tracking sequences. *IEEE Trans Image Process* Jun;2008 17(6):847–856. [PubMed: 18482881]
48. Risfic, B.; Arulampalam, S.; Gord, N. *Beyond the Kalman Filter: Particle Filters for Tracking Applications*. Boston, MA: Artech House; 2004.

Biographies

Oleg Michailovich (M'02) was born in Saratov, Russia, in 1972. He received the M.Sc. degree in electrical engineering from the Saratov State University in 1994 and the M.Sc. and Ph.D. degrees in biomedical engineering from The Technion—Israel Institute of Technology, Haifa, in 2003.

He is currently with the Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, ON, Canada. His research interests include the application of image processing to various problems of image reconstruction, segmentation, fusion, inverse problems, nonparametric estimations, approximation theory, and multiresolution analysis.

Allen Tannenbaum (M'93) was born in New York in 1953. He received the Ph.D. degree in mathematics from Harvard University, Cambridge, MA, in 1976.

He has held faculty positions at the Weizmann Institute of Science (Rehovot, Israel), McGill University, ETH Zurich, Technion, Ben-Gurion University of the Negev, and the University of Minnesota. He is presently the Julian Hightower Professor of Electrical and Biomedical Engineering at the Georgia Institute of Technology and Emory University, Atlanta, GA. He has done research in image processing, medical imaging, computer vision, robust control, systems theory, robotics, semiconductor process control, operator theory, functional analysis, cryptography, algebraic geometry, and invariant theory.

Fig. 1.
(Left panel) Simulated “aircraft” object; (right panel) simulated background.

Fig. 2.
 (Upper row of subplots) Simulated tracking images used in the computer-study; (middle row of subplots) segmentation masks obtained using “static Bayesian”; (lower row of subplots) segmentation masks obtained using “dynamic Bayesian.”

Fig. 3. Mean-squared error (MSE) of estimation of the “aircrafts” obtained with “static Bayesian” versus the MSE obtained with “dynamic Bayesian” for SNR = 20 dB.

Fig. 4.
 (Subplots A1–A5) Segmentation masks obtained using “dynamic Bayesian” for SNR = 15 dB;
 (subplots B1–B5) segmentation masks obtained using “dynamic Bayesian” for SNR = 10 dB.

Fig. 5. Subset of images forming one of the tracking sequences used in the experimental study of Section V-B. The images number 15, 30, 45, 60, 75, 90, 105, 120, 135, 150, 165, 180 (if counted in a normal left-to-right, up-to-down direction) are shown.

Fig. 6.
Segmentation of the images of Fig. 5 obtained using the “dynamic Bayesian” approach.

Fig. 7.
Segmentation of the images of Fig. 5 obtained using the motion-based segmentation approach of [17].

Fig. 8.

(Left subplot) Frame #150 of the tracking sequence in Fig. 5; (middle subplot) image segmentation using the motion-based segmentation approach of [17]; (right subplot) Image segmentation using the “dynamic Bayesian” approach.