Multiframe Detector/Tracker: Optimal Performance

MARCELO G. S. BRUNO, Member, IEEE
University of São Paulo
Brazil

JOSÉ M. F. MOURA, Fellow, IEEE
Carnegie Mellon University

We develop the optimal Bayes multiframe detector/tracker for rigid extended targets that move randomly in clutter. The performance of this optimal algorithm provides a bound on the performance of any other suboptimal detector/tracker. We determine by Monte Carlo simulations the optimal performance under a variety of scenarios including spatially correlated Gaussian clutter and non-Gaussian (K and Weibull) clutter. We show that, for similar tracking performance, the optimal Bayes tracker can achieve peak signal-to-noise ratio gains possibly larger than 10 dB over the commonly used combination of a spatial matched filter (spatial correlator) and a linearized Kalman–Bucy tracker. Simulations using real clutter data with a simulated target suggest similar performance gains when the clutter model parameters are unknown and estimated from the measurements.

Manuscript received September 19, 1999; revised August 22, 2000 and February 5, 2001; released for publication February 28, 2001.

IEEE Log No. T-AES/37/3/08563.

Refereeing of this contribution was handled by P. K. Willett.

M. G. S. Bruno was partially supported by CNPq-Brazil and FAPESP, São Paulo, Brazil. This work was also funded by ONR grant N00014-97-1-0800.

Authors' addresses: M. G. S. Bruno, Electrical Engineering Department, University of São Paulo, São Paulo, SP 05508-090, Brazil; J. M. F. Moura, Department of Electrical and Computer Engineering, Carnegie Mellon University, Pittsburgh, PA, 15213, E-mail: (moura@ece.cmu.edu).

I. INTRODUCTION

This work studies integrated detection and tracking of randomly moving targets in clutter using as input data a sequence of noisy images. The images may be collected by electromagnetic sensors such as high resolution radars, or optical sensors such as infrared (IR) devices. At each sensor scan, an image or frame is produced. If one or more targets are present during a scan, the corresponding image contains the returns from the targets plus the returns from the background clutter. Otherwise, if no target is present, the sensor return consists exclusively of clutter. The clutter accounts for spurious reflectors, which may appear as false targets, and for measurement noise.

In the case when multiple, at most M, targets of interest are present, the multitarget detector decides from the noisy data how many targets $(0,1,2,\ldots,M)$ are present in each frame. Once a target is declared present by the detector, a subsequent tracker estimates its position in the surveillance space. The interpolation across successive scans of the estimated positions of a target forms a track for that target. Due to clutter, false detections, known as false alarms, may occur, and false tracks may be estimated. Conversely, actual targets may fail to be detected. This situation is known as a miss. Even if correct detections (i.e., no misses or false alarms) occur, the background clutter can still cause the tracker to produce a wrong estimate of the position of the target, i.e., a tracking error.

The ultimate goal is to estimate the target state, typically a collection of kinematic components such as position, velocity, or acceleration. In most existing algorithms, e.g., [1], detection and tracking are two separate stages. The measurements of interest to the tracker are not the raw sensor images, but the outputs of preliminary detection subsystems. The detection stage involves the thresholding of the raw data, usually one single sensor frame. After further preprocessing, validated detections provide measurements that, for targets that are declared present, are treated as noise-corrupted observations of the target state such as, for example, direct estimates of position (range, azimuth, and elevation). Due to the ocurrence of random false alarms in the detection process, or due to clutter coming from spurious reflectors, interfering targets, or man-made decoys, validated measurements may actually be false measurements that do not originate from true targets.

Multitarget trackers generally assume [1, 8] that the targets are pointwise and associate a linear (or linearized) dynamic model to the state of each target of interest. A tracking filter, usually a variation on the Kalman–Bucy filter, combines the validated measurements with the dynamic model, providing an estimate of the state of the target. An important issue arising from the decoupling of detection and tracking is the problem of deciding which set of measurements

or weighted combination of measurements should be associated to each target state estimator or to clutter. This problem is known as data association. The most common data association algorithms, see [8], compute posterior probabilities of association conditioned on the measurements and use them throughout the estimation process.

Brief Review of the Literature: References concerned only with target detection, not tracking, include [4–6]. In [5], Pohlig introduces an algorithm for detection of constant velocity objects such as meteors, asteroids, and satellites, in fixed stellar backgrounds. The measurements are obtained by a staring sensor with an array of charged coupled device (CCD) sensors in the focal plane of a telescope. The focal plane image is integrated and sampled in space and time, resulting in a three-dimensional (two spatial dimensions and one temporal dimension) discrete model, where the optical intensity of both targets and the background are modeled as Poisson distributions with different means that reflect the different photon counts arising from targets and clutter. Pixel intensities under both hypotheses of presence and absence of target are assumed spatially uncorrelated. The detection algorithm in [5] is a 3D generalized likelihood ratio test (GLRT) based on batch processing: all available sensor frames are stacked in a data volume, and then the GLRT decides on the presence or absence of a target anywhere in that volume.

The work by Reed, Gagliardi, and Shao [6] is similar in nature to Pohlig's approach and introduces a 3D (again space plus time) matched filter for detection of known, moving targets within a Gaussian background clutter with known spectral density. However, unlike [5], [6] considers the case of continuous (nonsampled) data; it is best suited for optical rather than digital processing. A different problem is considered by Chen and Reed in [4]. The goal in [4] is to introduce a constant false alarm rate (CFAR) algorithm to solve the problem of detection of a known target signal in a given scene, using a set of K correlated reference scenes that contain no target or, alternatively, very weak target returns. The reference scenes are obtained either from different frequency bands of the main scene (multispectral or hyperspectral imagery) or from sequential observations in time. The proposed detection algorithm is a GLRT that tests for the presence or absence of a target in the main scene using as data the entire collection of reference scenes plus the main scene itself. The underlying model assumes that, after preprocessing (essentially removal of the local variable mean), each individual scene is a zero-mean, Gaussian, white random vector, i.e., the spatial correlation between the pixels in each individual image is neglected. However, the model assumes a cross-correlation between pixels at the same spatial location in different scenes. An alternative modeling for multispectral imagery that incorporates both interframe and intraframe correlation was proposed in [7].

Instead of decoupling detection and tracking as in [1], or considering detection-only of moving objects as in [5, 6], we develop the optimal, multiframe, Bayes detector/tracker that processes directly the sensor images and integrates detection and tracking into a unified framework. The Bayesian strategy involves the computation at each scan of the posterior probability of the unknown target states conditioned on the observations. In [3], the author uses a dynamic programing approach and the Viterbi algorithm to study target detection. We postpone to section IIIE a detailed discussion comparing the Bayes algorithm with the dynamic programming approach in [3].

In our approach, we integrate detection and tracking into the same framework by augmenting the target state space with additional dummy states that represent the absence of targets. The posterior probability of a given target being absent is propagated in time together with the posterior probabilities of the other "present target" states. In contrast to Pohlig's *batch* detector [5], we develop a *recursive* framework where we still process all frames available in an optimal way, but these frames are processed one by one and discarded as we finish processing them. As a new frame is available, we simply update the posterior probabilities for the target states by running one more iteration of the algorithm.

Modeling Assumptions: The optimal Bayesian algorithm takes full advantage of all prior information on the clutter, target signature, and target motion models, and allows multiframe detection and tracking with recursive processing across all observed sensor scans. We consider both pointwise (single pixel) and extended (multipixel) targets. We present detection results for targets with deterministic signatures and for targets with time-varying random signatures. The random signatures are described by multivariate, spatially correlated Gaussian distributions. We assume translational motions, and we define as the target state the spatial coordinates of the target's geometric centroid. Since practical sensors have a finite resolution, we restrict the target centroid positions to a finite grid where each pixel represents a resolution cell of the sensor. We describe motions by finite state machines (FSMs) obtained by discretizing the continuous differential equations that describe the target dynamics. The dummy states that represent the absence of a target are incorporated into the FSM model that also specifies the transition probabilities between the absence and the presence of a target, and vice-versa.

We consider two classes of clutter models: spatially correlated clutter with Gaussian statistics,

and uncorrelated non-Gaussian clutter with heavy tail amplitude (envelope) statistics. The spatial correlation of the clutter is captured by using noncausal, spatially homogeneous, Gauss–Markov random fields (GMRFs) of arbitrary order [25]. GMRFs are statistical models that capture the locality properties of the clutter, namely, the clutter at a given spatial location is strongly dependent on the clutter intensity in neighboring locations. This assumption is intuitively realistic in many practical scenarios. Regarding non-Gaussian clutter, we represent it by spherically invariant-random vectors (SIRVs) [13–15], which have been shown to generate a variety of envelope statistics of practical interest, including the Weibull, K, Rician [18], and G [19] envelopes.

Performance Studies: This work focuses on performance results for the optimal multiframe Bayes detector/tracker in a variety of scenarios, including, as we mentioned before, both deterministic and random signature targets, observed in both Gaussian and non-Gaussian clutter. We test the proposed algorithm primarily on synthetic data with known clutter and target models. The optimal performance curves, obtained through extensive Monte Carlo simulations, provide an upper bound to the performance of suboptimal algorithms. We benchmark against these bounds the performance of competing suboptimal schemes such as the association of a single frame spatial correlator (matched filter) with a multiframe linearized Kalman–Bucy filter (KBF) tracker. These studies show that there is a significant margin of improvement to be had over existing detectors and trackers.

In practice, the situation of perfect match between the data and the model is not realistic. In order to assess the robustness of the algorithm to mismatches between the measurements and the model, we present an example of detection/tracking with real clutter data, obtained by a laser radar mounted to the bottom of an aircraft. We fit the model to the real clutter by estimating its parameters from the data. The experimental results confirm that there is a significant improvement in performance over conventional algorithms such as a plain single frame image correlator associated with a KBF.

Summary: Section I is this Introduction. Section II presents the models for sensor, target, motion, and clutter that underly our integrated approach to detection and tracking. Section III examines the derivation of the optimal Bayesian detector/tracker based on the models from Section II. Sections IV and V quantify, respectively, the detection and tracking performances of the algorithm through comprehensive Monte Carlo simulations assuming a single target scenario. Both correlated Gaussian clutter and non-Gaussian clutter situations are considered, and performance comparisons with alternative suboptimal detection and tracking algorithms are provided. Finally, Section VI summarizes the contributions of this work.

We omit in this paper specific details on the implementation of the Bayes detector/tracker. These can be found in reference [24] for the particular case of a single, deterministic 2D target observed in GMRF clutter.

II. THE MODEL

At each sensor scan, there are at most M targets present in the surveillance space. Each target is a rigid body with translational motion belonging to one of M possible classes characterized by their signature parameters, and by the dimensions of their noise-free image. We restrict our discussion to the situation where all targets are distinct. For simplicity of notation, we restrict this section to one-dimensional (1D) spaces. A brief discussion on the corresponding 2D models and a comprehensive investigation of 2D detection/tracking performance are found in Section V (see also [24] for further details on modeling and implementation of the 2D detector/tracker algorithm).

A. Surveillance Space and Target Model

We first model the surveillance space of the sensor. Given the sensor's finite resolution, we discretize the 1D space by the uniform finite discrete lattice

$$\mathcal{L} = \{l : 1 \leq l \leq L\} \tag{1}$$

where L is the number of resolution cells and l is an integer. We refer to the lattice $\mathcal{L}$ as the *sensor lattice*. The resolution cells are also referred to as pixels.

To develop an integrated framework for detection and tracking, it is useful to extend the lattice $\mathcal{L}$ with additional states that will be used to represent the absence of targets and to account for the fact that target images extend over more than one pixel in the sensor lattice. We introduce first the vector $\mathbf{Z}_n = [z_n^1 \ldots z_n^M]^T$, which collects the positions of the geometric centroids of the M possible targets in the sensor image at scan n.

Let the pixel length S^p of a class p target, $1 \leq p \leq M$, be $S^p = (l_i^p + l_s^p + 1)$, where l_i^p and l_s^p are the maximum extent in pixels of the target, respectively to the left and to the right of its centroid. These parameters are assumed known and time invariant. If S^p is odd, we make $l_i^p = l_s^p = (S^p - 1)/2$. Otherwise, if S^p is even, we adopt the convention that $l_s^p = S^p/2$ and $l_i^p = l_s^p - 1$. To account for the situations when targets move in and out of the sensor range, we define the extended *centroid lattice*,

$$\bar{\mathcal{L}}_p = \{l : -l_s^p + 1 \leq l \leq L + l_i^p\} \tag{2}$$

which corresponds to the set of all possible centroid positions z_n^p such that at least one pixel of the target is still visible in the sensor image. Finally, to include

the possibility of absence of a target, we introduce an additional dummy state. We adopt the convention that, whenever a class p target is not present at the nth scan, z_n^p takes the value $L + l_i^p + 1$. With the addition of this dummy absent target state, we define the *augmented lattice*,

$$\tilde{\mathcal{L}}_p = \{l: -l_s^p + 1 \le l \le L + l_i^p + 1\}. \tag{3}$$

Extended Target Model: When a class p target is present, i.e., $z_n^p \in \bar{\mathcal{L}}_p$, the noise-free target image is simply the spatial distribution of the target pixel intensities, a_k^p, $-l_i^p \le k \le l_s^p$, centered at the centroid lattice cell z_n^p. Otherwise, if $z_n^p = L + l_i^p + 1$, meaning the target is absent, the sensor image corresponding to target-only returns reduces to a null image. These intuitive ideas are formalized mathematically by expressing the noise free image of a class p target at the nth sensor scan as the nonlinear function

$$\mathbf{t}_p(z_n^p) = \sum_{k=-l_i^p}^{l_s^p} a_k^p \mathbf{e}_{z_n^p + k} \qquad z_n^p \in \bar{\mathcal{L}}_p \tag{4}$$

$$\mathbf{t}_p(z_n^p) = \mathbf{0}_L \qquad z_n^p = L + l_i^p + 1 \tag{5}$$

where $\mathbf{e}_l$, $1 \le l \le L$, is an L-dimensional vector whose entries are all zero, except for the lth entry which is one. If $l < 1$ or $l > L$, $\mathbf{e}_l$ is defined as the identically zero vector. This particular definition for $\mathbf{e}_l$ outside the original sensor grid $\mathcal{L}$ is adopted to guarantee that the target model in (4) will accurately describe the disappearance of portions of the target from the sensor image as the target's centroid moves closer to the boundaries of the surveillance space.

The pixel intensity coefficients a_k^p in (4) are also known as the *target signature* coefficients. They may be deterministic and known, deterministic and unknown, or random. Random signatures account for fluctuations in the reflectivity, or in the conditions of illumination of the target, as well as random variations in channel characteristics such as fading. For simplicity, we assume in most of this paper that the signature coefficients are known and time invariant. An extension of the detection/tracking algorithms to targets with random signature is presented in Section IIID. Monte Carlo simulations with synthetic spatially correlated/temporally uncorrelated Gaussian targets are presented in Section IV.

B. Multitarget Observations and Clutter Models

We consider a multitarget scenario with M possible targets, and collect the L sensor readings at each pixel of the nth scan in the L-dimensional column vector $\mathbf{y}_n$. Due to the presence of spurious reflectors and background, $\mathbf{y}_n$ consists of the superposition of the various noise-free target images plus clutter. Using the extended target model introduced in Subsection IIA, the observation vector or nth sensor frame $\mathbf{y}_n$ is given by

$$\mathbf{y}_n = \mathbf{t}_1(z_n^1) + \mathbf{t}_2(z_n^2) + \cdots + \mathbf{t}_M(z_n^M) + \mathbf{v}_n \tag{6}$$

where $\mathbf{v}_n$ is the background clutter vector, also referred to as the nth clutter frame, and $\mathbf{t}_p(z_n^p)$, $1 \le p \le M$, is given by (4) and (5), depending on whether the pth target is present or absent at the nth scan. The clutter $\mathbf{v}_n$ is assumed to be statistically independent of $\mathbf{t}_p(z_n)$, $1 \le p \le M$.

At each frame, the clutter at a given spatial location (pixel) may be statistically correlated with the clutter at another spatial location. The clutter intensity may also have Gaussian or non-Gaussian statistics. We adopt one of three models for $\mathbf{v}_n$: spatially white Gaussian clutter; spatially correlated Gaussian clutter; and spatially white non-Gaussian clutter. These models allow us to assess how clutter spatial correlation or non-Gaussian clutter statistics affect the performance of the detection/tracking algorithms.

Gaussian Clutter: Under the assumption of Gaussianity, the vector $\mathbf{v}_n$ has a multivariate normal probability density function (pdf), $p(\mathbf{v}_n) = N(\mathbf{0}, \mathbf{R})$, where $\mathbf{R}$ is the clutter spatial covariance, and $\mathbf{0}$ is the mean. The zero-mean assumption assumes a preprocessing stage that removes the possibly spatially variant local mean. A non-zero mean can be accounted for trivially. We distinguish two cases for the covariance matrix $\mathbf{R}$.

White spatially homogeneous Gauss clutter: With spatially uncorrelated (white) clutter, the covariance matrix $\mathbf{R}$ is diagonal. Assuming spatial homogeneity, $\mathbf{R} = \sigma_v^2 \mathbf{I}$, where $\mathbf{I}$ is the identity matrix and σ_v^2 is the variance (power) of the clutter.

Spatially correlated homogeneous Gauss–Markov clutter: We model spatially correlated clutter as a GMRF [25]. This model simply states that the clutter intensity at a given pixel of the sensor image is a weighted average of the clutter intensity in neighboring pixels plus an error term. We assume here a *noncausal* neighborhood region for each pixel. If we add the assumption of spatial homogeneity, an mth order 1D noncausal GMRF model for the nth clutter frame is given by the spatial difference equation

$$v_n(l) = \sum_{j=1}^{m} \alpha_j [v_n(l-j) + v_n(l+j)] + u_n(l) \qquad 1 \le l \le L \tag{7}$$

where $u_n(l)$ is a zero-mean, correlated prediction error such that

$$E[v_n(l) u_n(k)] = 0 \qquad \forall \quad k \ne l \tag{8}$$

and the symbol $E[.]$ stands for expectation or ensemble average. In order to completely define (7) at all pixel locations, we specify boundary conditions (BCs) outside the sensor lattice $\mathcal{L}$. Common boundary conditions are simply $v_n(l) = 0$ for $l < 1$ or $l > L$.

These are known as Dirichlet BCs. Other BCs can be alternatively used, see for example [23, 25].

Second-order statistics of GMRFs: The GMRF model is very attractive because it provides a simple parameterization for the inverse of the covariance matrix of the background clutter $\mathbf{v}_n$. Collecting the clutter samples $v_n(l)$ and the error samples $u_n(l)$, $1 \le l \le L$, in two L-dimensional vectors $\mathbf{v}_n$ and $\mathbf{u}_n$, an equivalent matrix representation for the difference equation in (7) is

$$\mathbf{A}\mathbf{v}_n = \mathbf{u}_n \tag{9}$$

where $\mathbf{A}$ is a sparse and highly structured matrix usually referred to as the *potential matrix*. For the 1D mth order homogeneous model in (7), the potential matrix is an m-banded, Toeplitz, symmetric matrix with structure [25]

$$\mathbf{A} = \begin{bmatrix} 1 & -\alpha_1 & -\alpha_2 & \cdots & -\alpha_m & 0 & \cdots & \cdots & \cdots & \cdots & \cdots & 0 \\ -\alpha_1 & 1 & -\alpha_1 & -\alpha_2 & \cdots & -\alpha_m & 0 & \cdots & \cdots & \cdots & \cdots & 0 \\ -\alpha_2 & -\alpha_1 & 1 & -\alpha_1 & \cdots & \cdots & -\alpha_m & 0 & \cdots & \cdots & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & 0 & \cdots & \cdots & 0 \\ -\alpha_m & \cdots & \cdots & \cdots & 1 & -\alpha_1 & -\alpha_2 & \cdots & -\alpha_m & 0 & \cdots & 0 \\ 0 & -\alpha_m & \cdots & \cdots & \cdots & 1 & -\alpha_1 & \cdots & \cdots & -\alpha_m & \cdots & 0 \\ \mathbf{0} & \mathbf{0} & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \mathbf{0} \\ 0 & \cdots & \cdots & -\alpha_m & \cdots & \cdots & \cdots & 1 & \cdots & \cdots & \cdots & -\alpha_m \\ 0 & \cdots & \cdots & 0 & \alpha_m & \cdots & \cdots & -\alpha_1 & 1 & \cdots & \cdots & -\alpha_{m-1} \\ \mathbf{0} & \vdots & \vdots & \vdots & \mathbf{0} & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \mathbf{0} & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & \cdots & \cdots & \cdots & \cdots & 0 & -\alpha_m & \cdots & \cdots & -\alpha_1 & 1 \end{bmatrix}. \tag{10}$$

We now derive the second-order statistics of $\mathbf{u}_n$, which is referred to as the prediction error, and of the clutter field $\mathbf{v}_n$. Combining the orthogonality condition in (8) with the matrix equation (9), we note that

$$E[\mathbf{u}_n\mathbf{u}_n^T] = E[\mathbf{A}\mathbf{v}_n\mathbf{u}_n^T] = \sigma_u^2\mathbf{A} \tag{11}$$

where ^{T} denotes the transpose of a vector or matrix. In (11), we used the assumption of spatial homogeneity (roughly speaking, the spatial "equivalent" of stationarity) to make $E[v_n(l)u_n(l)] = \sigma_u^2$, for all l, $1 \le l \le L$.

Finally, since $\mathbf{A}$ is nonsingular and symmetric, then $\mathbf{v}_n = \mathbf{A}^{-1}\mathbf{u}_n$ and

$$\begin{aligned} \Sigma_v^{-1} &= (E[\mathbf{v}_n\mathbf{v}_n^T])^{-1} = (\mathbf{A}^{-1}E[\mathbf{u}_n\mathbf{u}_n^T]\mathbf{A}^{-T})^{-1} \\ &= (\sigma_u^2\mathbf{A}^{-1}\mathbf{A}\mathbf{A}^{-T})^{-1} = \mathbf{A}/\sigma_u^2. \end{aligned} \tag{12}$$

Equation (12) gives for free, with no matrix inversion required, the inverse of the clutter covariance in terms of the highly structured matrix $\mathbf{A}$ given in (10). This structure is used to design computationally efficient detection and tracking algorithms when the clutter is correlated as a GMRF. Finally, for our simulation studies, we use equation (9) and a technique based on the upper Cholesky factorization of the potential matrix $\mathbf{A}$, [25], to generate samples of the GMRF clutter $\mathbf{v}_n$.

Non-Gaussian Clutter: When dealing with non-Gaussian clutter, we assume that the sensor measures, at each resolution cell, the in-phase and quadrature returns of the clutter and targets echoes. The clutter measurements at instant n correspond to a sampling of the returned clutter complex envelope and are given by the even-sized vector

$$\mathbf{v}_n = [v_{c_n}^1 \quad v_{s_n}^1 \ldots v_{c_n}^L v_{s_n}^L]^T \tag{13}$$

where L is the number of resolution cells. We assume that the double-sized vector $\mathbf{v}_n$ has a joint pdf with *non-Gaussian* statistics such that the sequence of random variables

$$e_k = \sqrt{(v_{c_n}^k)^2 + (v_{s_n}^k)^2} \qquad 1 \le k \le L \tag{14}$$

is identically distributed with a pdf different from a Rayleigh distribution.

K and Weibull envelope statistics: We are interested in analyzing how the tracker performs against a background clutter whose envelope at each resolution cell has heavier tails than a Rayleigh envelope. Useful clutter envelope statistics are the K and Weibull models that are frequently used in the literature to represent the amplitude statistics of clutter returns [20–22]. The corresponding pdfs for the two models are [18] as follows.

1) *K pdf*: $p_E(e) = b^{\nu+1}e^{\nu}/2^{\nu-1}\Gamma(\nu)K_{\nu-1}(be)\ e \geq 0$ where ν is a shape parameter, $\Gamma(.)$ is the Eulerian function, $K_{\nu-1}(.)$ is a modified Bessel function of the second kind and b is related to σ^2 by $b^2 = (2\nu)/\sigma^2$.

2) *Weibull pdf*: $p_E(e) = ace^{c-1}\exp(-ae^c)\ e \geq 0$ where c is a shape parameter and a relates to the average power σ of the quadrature components by $2\sigma^2 = a^{-2/c}\Gamma(1 + 2/c)$.

Simulation of K and Weibull clutter samples: Rayleigh envelope statistics correspond to a multivariate joint Gaussian distribution of the in-phase and quadrature clutter returns. Similarly, heavy-tailed envelope statistics such as the Weibull and K distributions correspond to a generalized SIRV model in the backscatter domain [13, 14]. Techniques to simulate heavy-tailed clutter using SIRV models have been discussed extensively in the literature [14–16]. In particular, we used the algorithms in [16] to generate the samples of uncorrelated K and Weibull clutter that were used in the Monte Carlo simulations in Section IVB. We omit the simulation details here for lack of space and refer the reader instead to the literature, particularly [16].

C. Target Motion

Assuming that the targets are rigid bodies with translational motion, the target motion is completely specified by the dynamics of the target centroid. We adopt a first-order statistical model for the centroid dynamics. Given the sensor finite resolution, we model the motion of a class p target in the corresponding augmented lattice $\tilde{\mathcal{L}}_p$ by a set of transition probabilities

$$\{P(z_n^p = k \mid z_{n-1}^p = j)\} \qquad k, j \in \tilde{\mathcal{L}}_p. \tag{15}$$

The transition probabilities $P(z_n^p \mid z_{n-1}^p)$, $1 \leq p \leq M$, represent the likelihood of displacement of a class p target between two consecutive frames. The transition probabilities in (15) define an FSM that specifies the dynamics of the centroid for the class p target.

Example: Targets with constant mean velocity: A particular dynamic model of interest is a target where the nominal velocity is constant. We perturb this mean or nominal velocity by an mth-order random walk fluctuation. The target centroid position at instant n is then given by

$$z_{n+1} = z_n + d + \varepsilon_n \tag{16}$$

where d is the mean velocity, and ε_n is a discrete-valued, zero-mean white noise component that is independent of the centroid position and takes values on the discrete set $\mathcal{S} = \{-m, \ldots -1, 0, 1, \ldots, m\}$ for some $m \geq 1$. Fig. 1 shows the central section of the FSM that corresponds to the model in (16) when $d = 0$ and $m = 1$. This FSM is simply a first-order discrete Markov chain. As mentioned before, a target that is present moves to the absent state whenever its centroid is outside the lattice $\tilde{\mathcal{L}}_p = \{l : -l_s^p + 1 \leq l \leq L + l_i^p\}$. When no target of a given class is present, we assume that there is a non-zero probability p_a of a new target from that same class appearing randomly at the next sensor scan. We assume that the target centroid may appear at any pixel of the centroid lattice $\tilde{\mathcal{L}}_p$, with equal probability $p_a/(L + l_i^p + l_s^p)$. This assumption is a *worst case* scenario, when the detector/tracker has no a priori information about initial position of a new target. Other more elaborate distributions for the probability of reappearance are easily taken into account.

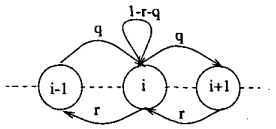


Fig. 1. Example of FSM diagram.

III. OPTIMAL BAYES MULTITARGET DETECTOR/TRACKER

We assume that at each scan n an unknown number of targets ranging from zero to M may be present. The targets that are present belong to distinct classes (i.e., in the context of this model, they have different signatures). We collect the observation scans from instant 0 up to instant n in the long observation vector $\mathbf{Y}_0^n = [\mathbf{y}_0^T \ldots \mathbf{y}_n^T]^T$. Given $\mathbf{Y}_0^n$, we want to perform three tasks at instant n: 1) determine how many targets are present/absent (detection); 2) assign the detected targets to a given class (data association); 3) estimate the positions of the detected targets (tracking).

A. Nonlinear Stochastic Filtering Approach

As mentioned in Section II, the vector

$$\mathbf{Z}_n = [z_n^1 \ldots z_n^M]^T \tag{17}$$

collects the positions of the centroids of the M possible targets in the sensor image. If all $z_n^p = L + l_i^p + 1$, $1 \leq p \leq M$, then no target is present in the surveillance space at the nth scan. The optimal Bayes solution to the joint detection/tracking problem is obtained by computing at each scan the joint posterior probability, $P(\mathbf{Z}_n \mid \mathbf{Y}_0^n)$, i.e., the conditional probability of the vector and centroid positions.

The Bayes detector/tracker that we present processes the observations as they become available. It computes recursively $P(\mathbf{Z}_n \mid \mathbf{Y}_0^n)$ at each scan, thus avoiding having to store all the measurements from instant zero up to the present. The recursion is divided into two steps. The first step is the prediction step. It uses the statistical description of the target motion between two consecutive scanned frames to predict the current position of the targets based on all past observations. Once a new sensor frame is available, a

second step, known as the filtering step, uses the new measurements to correct the prediction. The incoming sensor data is processed using the information in the clutter and target signature models. In the sequel, we describe both steps in further detail.

The following assumptions are made in the derivation of the algorithm.

1) In each frame, only one target from each of the M possible classes may be present.

2) The sequence of clutter frames $\{\mathbf{v}_n\}$, $n \geq 1$, is independent, identically distributed (iid).

3) The sequence of target states $\{\mathbf{Z}_k\}$, $k \geq 0$, is statistically independent of the sequence of clutter frames $\{\mathbf{v}_k\}$, $k \geq 0$.

4) Targets from different classes move independently and the translational motions for targets from each class are described by first-order discrete Markov Chains completely specified by the transition probabilities $P(z_n^p \mid z_{n-1}^p)$, $1 \leq p \leq M$, $z_n^p \in \tilde{\mathcal{L}}_p$.

5) In all observed frames, the target signatures are deterministic and known (but not necessarily time-invariant) for each target class.

We make the following remarks regarding the previous assumptions:

a) The detector/tracker algorithm can be easily modified to account for unknown, random target signatures in each sensor frame. We discuss the necessary modifications in Subsection IIID.

b) Instead of assuming that at most one target from each class is present in each frame, we could have used an alternative problem setup in which there is a *known* maximum number of targets, $\mathcal{N}_p \geq 1$, for each target class p. In this paper, for convenience, and without loss of generality, we make $\mathcal{N}_p = 1$, $\forall p$, $1 \leq p \leq M$.

c) The assumption that the sequence $\{\mathbf{v}_n\}$, $n \geq 0$, is iid is equivalent to ignoring all interframe statistical correlation between the clutter pixels. The 2D GMRF model in Section IIB assumes however an intraframe or spatial clutter correlation.

We now detail the derivation of the algorithm. In the subsequent derivation, we denote the probability mass function of discrete-valued random variables by the capital letter P, whereas the pdf of continuous-valued random variables is denoted by lowercase p.

Prediction Step: This step computes the *prediction posterior probability*

$$P(\mathbf{Z}_n \mid \mathbf{Y}_0^{n-1}) = P(z_n^1, \ldots, \ldots z_n^M \mid \mathbf{Y}_0^{n-1}) \quad z_n^p \in \tilde{\mathcal{L}}_p \quad 1 \leq p \leq M. \tag{18}$$

From $P(\mathbf{Z}_n \mid \mathbf{Y}_0^{n-1})$, we can obtain the marginal posterior probabilities of the centroid position of each target conditioned on the past frames from instant 0 to instant $n-1$. We also obtain the posterior probabilities of absence of each target conditioned on the past observations.

Combining the Theorem of Total Probability with Bayes law, we write

$$\begin{aligned} P(\mathbf{Z}_n \mid \mathbf{Y}_0^{n-1}) &= \sum_{\mathbf{Z}_{n-1}} P(\mathbf{Z}_n, \mathbf{Z}_{n-1} \mid \mathbf{Y}_0^{n-1}) \\ &= \sum_{\mathbf{Z}_{n-1}} P(\mathbf{Z}_n \mid \mathbf{Z}_{n-1}, \mathbf{Y}_0^{n-1}) P(\mathbf{Z}_{n-1} \mid \mathbf{Y}_0^{n-1}). \end{aligned} \tag{19}$$

Since the sequence of target centroid positions $\{\mathbf{Z}_k\}$, $k \geq 1$, is, by assumption, a first-order Markov process, then, conditioned on $\mathbf{Z}_{n-1}$, the current state $\mathbf{Z}_n$ is statistically independent of the sequence $\{\mathbf{Z}_k\}$, $0 \leq k \leq n-2$. If we add the assumption that $\mathbf{Z}_n$ is also independent of the sequence of previous clutter frames $\{\mathbf{v}_k\}$, $0 \leq k \leq n-1$, $n \geq 1$, we conclude that, conditioned on $\mathbf{Z}_{n-1}$, $\mathbf{Z}_n$ is statistically independent of the previous observations, $\mathbf{Y}_0^{n-1}$, i.e.,

$$P(\mathbf{Z}_n \mid \mathbf{Z}_{n-1}, \mathbf{Y}_0^{n-1}) = P(\mathbf{Z}_n \mid \mathbf{Z}_{n-1}). \tag{20}$$

Replacing (20) in (19), we get

$$P(\mathbf{Z}_n \mid \mathbf{Y}_0^{n-1}) = \sum_{\mathbf{Z}_{n-1}} P(\mathbf{Z}_n \mid \mathbf{Z}_{n-1}) P(\mathbf{Z}_{n-1} \mid \mathbf{Y}_0^{n-1}). \tag{21}$$

Finally, assuming that the different targets move according to statistically independent Markov chains,

$$P(\mathbf{Z}_n \mid \mathbf{Z}_{n-1}) = P(z_n^1 \mid z_{n-1}^1) \ldots P(z_n^M \mid z_{n-1}^M) \tag{22}$$

and we write

$$\begin{aligned} P(\mathbf{Z}_n \mid \mathbf{Y}_0^{n-1}) = \sum_{z_{n-1}^1} \cdots \sum_{z_{n-1}^M} & P(z_n^1 \mid z_{n-1}^1) \\ & \ldots P(z_n^M \mid z_{n-1}^M) P(\mathbf{Z}_{n-1} \mid \mathbf{Y}_0^{n-1}). \end{aligned} \tag{23}$$

Filtering Step: We now compute the *filtering posterior probability*, $P(\mathbf{Z}_n \mid \mathbf{Y}_0^n)$. From Bayes' law,

$$P(\mathbf{Z}_n \mid \mathbf{Y}_0^n) = P(\mathbf{Z}_n \mid \mathbf{y}_n, \mathbf{Y}_0^{n-1}) \tag{24}$$

$$= C_n p(\mathbf{y}_n \mid \mathbf{Z}_n, \mathbf{Y}_0^{n-1}) P(\mathbf{Z}_n \mid \mathbf{Y}_0^{n-1}) \tag{25}$$

$$= C_n p(\mathbf{y}_n \mid \mathbf{Z}_n) P(\mathbf{Z}_n \mid \mathbf{Y}_0^{n-1}) \tag{26}$$

where C_n is a normalization constant. To write equation (26), we used the fact that the sequence of clutter vectors $\{\mathbf{v}_n\}_{n=0}^{\infty}$ is iid and independent of the sequence of state vectors $\{\mathbf{Z}_n\}_{n=0}^{\infty}$. Hence, given $\mathbf{Z}_n$, vector $\mathbf{y}_n$ is independent of $\mathbf{Y}_0^{n-1}$. The term $p(\mathbf{y}_n \mid \mathbf{Z}_n)$ is referred to in the nonlinear stochastic filtering literature as the *observations kernel* [28, 30] and specifies the conditional statistics of the observed data assuming that the targets' states (positions) are known. The analytical expression for the observation kernel depends on the clutter and target models. We present next the optimal detection and tracking algorithms.

B. Minimum Probability of Error Bayes Detector

For each of the M possible targets, there are two possible detection states during the nth scan: absent or present. The detection algorithm is therefore a statistical test that, based on all present and past observed data, $\mathbf{Y}_0^n$, chooses one among 2^M possible hypotheses H_m, $0 \le m \le 2^M - 1$. In this notation, hypothesis H_0 stands for "all M possible targets are absent." Conversely, hypothesis H_{2^M-1} means that all M possible targets are present. Hypotheses H_m, $m \neq 0$ and $m \neq 2^M - 1$, represent all other combinations in between of presence and absence of the multiple targets.

Given $P(\mathbf{Z}_n \mid \mathbf{Y}_0^n)$, we compute the posterior probabilities of the detection hypothesis H_m, $0 \le m \le 2^M - 1$. The minimum probability of error detector decides that hypothesis H_m is true if [31]

$$P(H_m \mid \mathbf{Y}_0^n) > P(H_k \mid \mathbf{Y}_0^n) \quad \forall \quad k \neq m, \quad 0 \le m, \quad k \le 2^M - 1 \tag{27}$$

where $P(H_m \mid \mathbf{Y}_0^n)$ is the posterior probability of hypothesis H_m. We describe two illustrative examples.

Example 1: Single Target

With a single target, there are only two possible hypotheses at each sensor scan:

1) H_0: {target absent}.
2) H_1: {target present}.

The minimum probability of error detector assuming equal cost assignment to misses and false alarms and zero cost assignment to correct decisions reduces to

$$\frac{P(H_0 \mid \mathbf{Y}_0^n) \overset{H_0}{\gtrless} 1.}{P(H_1 \mid \mathbf{Y}_0^n) \underset{H_1}{}} \tag{28}$$

Introducing the posterior probability vector $\mathbf{f}_{n|n}$ such that its kth component is

$$f_{n|n}(k) = P(z_n^1 = k \mid \mathbf{Y}_0^n) \qquad k \in \tilde{\mathcal{L}}_1 \tag{29}$$

then

$$P(H_0 \mid \mathbf{Y}_0^n) = f_{n|n}(L + l_i^1 + 1) \tag{30}$$

$$P(H_1 \mid \mathbf{Y}_0^n) = 1 - f_{n|n}(L + l_i^1 + 1). \tag{31}$$

Remark. The decision rule in (28) minimizes the total probability of decision errors, misses, and false alarms. Alternatively, if we change the threshold in (28) and vary it over a wide range, the detection algorithm operates as a Neyman–Pearson detector [31] that maximizes the probability of detection for a given probability of false alarm.

Example 2: Two Targets

We illustrate next how to compute the quantities $P(H_m \mid \mathbf{Y}_0^n)$ from the filtering posterior probability $P(\mathbf{Z}_n \mid \mathbf{Y}_0^n)$ when there are two targets, i.e., $M = 2$. With two targets, there are 4 possible hypotheses for the presence or absence of targets at the nth sensor scan:

1) H_0 = {Both targets absent}.
2) H_1 = {Target 1 absent and Target 2 present}.
3) H_2 = {Target 1 present and Target 2 absent}.
4) H_3 = {Both targets present}.

We introduce the filtering posterior probability matrix $\mathbf{F}_{n|n}$ whose (k,j) element is the conditional probability that target 1 is at pixel k and target 2 is at pixel j, conditioned on the observation path $\mathbf{Y}_0^n$, i.e.,

$$F_{n|n}(k,j) = P(z_n^1 = k, z_n^2 = j \mid \mathbf{Y}_0^n) \qquad k \in \tilde{\mathcal{L}}_1, \quad j \in \tilde{\mathcal{L}}_2. \tag{32}$$

The posterior probabilities of the different hypothesis are computed as follows:

$$\begin{aligned} P(H_0 \mid \mathbf{Y}_0^n) &= F_{n|n}(L + l_i^1 + 1, L + l_i^2 + 1) \\ P(H_1 \mid \mathbf{Y}_0^n) &= \sum_{j=-l_s^2+1}^{L+l_i^2} F_{n|n}(L + l_i^1 + 1, j) \\ P(H_2 \mid \mathbf{Y}_0^n) &= \sum_{k=-l_s^1+1}^{L+l_i^1} F_{n|n}(k, L + l_i^2 + 1) \\ P(H_3 \mid \mathbf{Y}_0^n) &= \sum_{k=-l_s^1+1}^{L+l_i^1} \sum_{j=-l_s^2+1}^{L+l_i^2} F_{n|n}(k,j). \end{aligned} \tag{33}$$

The posterior probability of the two targets being present can be alternatively calculated as

$$P(H_3 \mid \mathbf{Y}_0^n) = 1 - \sum_{r=0}^{2} P(H_r \mid \mathbf{Y}_0^n). \tag{34}$$

C. Tracking: Maximum a Posteriori Tracker

We examine next the solution to the tracking (localization) problem. We use a *maximum a posteriori* (MAP) strategy that gives optimal localization in a Bayesian sense, with respect to a cost function that assigns uniform penalty to any tracking error regardless of the magnitude of the error [31].

If, after detection, hypothesis H_m, $1 \le m \le 2^M - 1$, is declared true, we introduce the conditional probability tensor $\mathbf{\Pi}_{n|n}^m$ defined as

$$\Pi_{n|n}^m(\mathbf{Z}_n) = P(\mathbf{Z}_n \mid H_m, \mathbf{Y}_0^n) = \frac{P(\mathbf{Z}_n, H_m \mid \mathbf{Y}_0^n)}{P(H_m \mid \mathbf{Y}_0^n)}. \tag{35}$$

The MAP Bayes tracker looks for the maximum of $\mathbf{\Pi}_{n|n}^m$ over $\mathbf{Z}_n$ to estimate the positions of the targets that are assumed present under hypothesis H_m.

Example 1: Single Target

In the single target case, the tensor $\Pi^1_{n|n}$ reduces to a vector whose general element is

$$\Pi^1_{n|n}(k) = P(z_n = k \mid \text{target is present}, \mathbf{Y}_0^n) = \frac{f_{n|n}(k)}{1 - f_{n|n}(L + l_i^1 + 1)}. \quad (36)$$

When the target is present, the MAP estimates of the actual target position are

$$\widehat{z^1_{n|n}} = \arg \max_{-l_s^1+1 \le k \le L+l_i^1} \Pi^1_{n|n}(k). \quad (37)$$

Example 2: Two Targets

In the case of two targets, the conditional probability tensors $\mathbf{\Pi}^k_{n|n}$, $k = 1,2,3$ are matrices. Let $\mathbf{F}_{n|n}$ be the filtering posterior probability matrix defined in (32) and let H_1, H_2, and H_3 be the three possible "target present" hypotheses as described before. We have three cases as follows.

1) Target 1 is declared absent and target 2 is declared present: In this case, we find the optimal MAP estimate of the centroid position of the class 2 target, denoted by $\widehat{z^2_{n|n}}$, using the expression

$$\widehat{z^2_{n|n}} = \arg \max_{j \in \bar{\mathcal{L}}_2} \frac{F_{n|n}(L + l_i^1 + 1, j)}{P(H_1 \mid \mathbf{Y}_0^n)}. \quad (38)$$

2) Target 1 is declared present and target 2 is declared absent: This situation is the dual of the previous case. The optimal MAP estimate of the centroid position of the class 1 target, denoted by $\widehat{z^1_{n|n}}$, is given by

$$\widehat{z^1_{n|n}} = \arg \max_{k \in \bar{\mathcal{L}}_1} \frac{F_{n|n}(k, L + l_i^2 + 1)}{P(H_2 \mid \mathbf{Y}_0^n)}. \quad (39)$$

3) Targets 1 and 2 are declared present: When both targets are declared present, the optimal MAP centroid estimates are

$$(\widehat{z^1_{n|n}}, \widehat{z^2_{n|n}}) = \arg \max_{k \in \bar{\mathcal{L}}_1, j \in \bar{\mathcal{L}}_2} \frac{F_{n|n}(k,j)}{P(H_3 \mid \mathbf{Y}_0^n)} \qquad k \in \bar{\mathcal{L}}_1, \quad j \in \bar{\mathcal{L}}_2. \quad (40)$$

D. Detection/Tracking of Targets with Random Signature

In the previous subsections, we considered the situation where the targets' signatures are deterministic and known. We now extend the algorithm to account for targets with random pixel intensity.

For simplicity, assume that the targets have equal size, i.e., $l_i^p = l_i$ and $l_s^p = l_s$, $1 \le p \le M$, where M is the number of targets. Let $\bar{l} = l_i + l_s + 1$ and define the $\bar{l}$-dimensional column vector of the signature parameters $\mathbf{a}_n^p$ such that its ith component is

$$a_n^p(i) = a_{i,n}^p \qquad -l_i \le i \le l_s \qquad 1 \le p \le M. \quad (41)$$

Now stack these signature parameters in the column vector

$$\mathbf{\Theta}_n = [(\mathbf{a}_n^1)^T \ldots (\mathbf{a}_n^M)^T]^T. \quad (42)$$

Assume that the sequence $\{\mathbf{\Theta}_n\}$ is iid and independent of $\{\mathbf{Z}_n\}$ and $\{\mathbf{v}_n\}$, for $n \ge 1$. After a few algebraic steps, it is easy to show that

$$P(\mathbf{Z}_n \mid \mathbf{Y}_0^n) = C_n \left[\int p(\mathbf{y}_n \mid \mathbf{\Theta}_n, \mathbf{Z}_n) p(\mathbf{\Theta}_n) d\mathbf{\Theta}_n \right] P(\mathbf{Z}_n \mid \mathbf{Y}_0^{n-1}). \quad (43)$$

Equation (43) shows that, under the assumptions that the sequence of target signatures $\{\mathbf{\Theta}_n\}$ is iid and statistically independent of both the sequence of target positions and the sequence of clutter frames, we can obtain the observations kernel for each possible state vector $\mathbf{Z}_n$ at instant n by averaging the conditional pdf of the measurements $p(\mathbf{y}_n|\mathbf{\Theta}_n, \mathbf{Z}_n)$ over all possible realizations of the vector of target signatures.

E. Comparison with Dynamic Programming Approaches

In this subsection, we contrast the nonlinear stochastic filtering approach to target tracking with previous work by Barniv [3]. We contrast Barniv's paper [3] with ours with respect to two issues: 1) the Viterbi algorithm used in [3] versus Bayes' law as used by us; 2) setup of the problem and other modeling assumptions. We also make some brief comments on computational complexity.

1) *Viterbi Algorithm versus Bayes' Law*: For simplicity of notation and to follow Barniv's model, we consider a single target scenario and assume initially that the target is present in the surveillance space in all observed sensor frames, so that no hard detection decision between presence or absence of target has to be made at each sensor scan. Reference [3] applies Bellman's dynamic programming [12] and its implementation by Viterbi [10] and Larson [11] to solve the target trajectory estimation problem. Let z_k and $\mathbf{y}_k$ denote, respectively, the unknown target state and the observations at instant k. Define the target state path $\mathbf{Z}_0^k$ and the observation path $\mathbf{Y}_0^k$ such that

$$\mathbf{Z}_0^k = [z_0, z_1, \ldots, z_k] \quad (44)$$

$$\mathbf{Y}_0^k = [\mathbf{y}_0, \mathbf{y}_1, \ldots \mathbf{y}_k]. \quad (45)$$

Denote by $\hat{z}_{k+1|k+1}$ the estimate of the unknown target state z_{k+1} at instant $k+1$, based on the observations $\mathbf{Y}_0^{k+1}$. Barring some minor differences in the indexing of the state and observation paths, Barniv's estimate

of the target state is given by

$$\hat{z}_{k+1|k+1} = \arg\max_{z_{k+1}} I(z_{k+1}) \tag{46}$$

where $I(z_k)$ is a quantity that is proportional to

$$\max_{\mathbf{Z}_0^{k-1}} P(\mathbf{Z}_0^k \mid \mathbf{Y}_0^k) \tag{47}$$

and is computed using the recursion

$$I(z_{k+1}) = p(\mathbf{y}_{k+1} \mid z_{k+1}) \max_{z_k} [P(z_{k+1} \mid z_k) I(z_k)] \qquad k \geq 0. \tag{48}$$

When the observation $\mathbf{y}_0$ at instant zero is available, (48) is initialized with

$$I(z_0) = p(\mathbf{y}_0 \mid \mathbf{z}_0) P(z_0). \tag{49}$$

Barring some minor differences in the initialization of the algorithm due to the availability of the observation $\mathbf{y}_0$, (48) corresponds essentially to the forward recursion step of the Viterbi algorithm, see [10]. By contrast, our tracking algorithm is a MAP estimator based on Bayes' law, i.e., our estimate for the unknown target state at instant $k+1$ is

$$\hat{z}_{k+1|k+1} = \arg\max_{z_{k+1}} P(z_{k+1} \mid \mathbf{Y}_0^{k+1}). \tag{50}$$

Combining the prediction and filtering steps of our algorithm in one equation, the posterior probability mass function (pmf) on the right-hand side of (50) is obtained by the recursion

$$P(z_{k+1} \mid \mathbf{Y}_0^{k+1}) = C_{k+1} p(\mathbf{y}_{k+1} \mid z_{k+1}) \left[\sum_{z_k} P(z_{k+1} \mid z_k) P(z_k \mid \mathbf{Y}_0^k) \right] \qquad k \geq 0 \tag{51}$$

where C_{k+1} is a normalization constant that is independent of z_{k+1}. We initialize (51) with

$$P(z_0 \mid \mathbf{y}_0) = C_0 p(\mathbf{y}_0 \mid z_0) P(z_0) \tag{52}$$

where C_0 is a normalization constant that is independent of z_0.

Equations (48) and (51) clearly define two different recursive algorithms. We now show that equations (46) and (50) correspond to two different maximization problems and may lead to different state estimates. Write

$$P(\mathbf{Z}_0^{k+1} \mid \mathbf{Y}_0^{k+1}) = P(\mathbf{Z}_0^k, z_{k+1} \mid \mathbf{Y}_0^{k+1}) = P(\mathbf{Z}_0^k \mid z_{k+1}, \mathbf{Y}_0^{k+1}) P(z_{k+1} \mid \mathbf{Y}_0^{k+1}). \tag{53}$$

The second factor in (53) is the conditional pmf of the current state z_{k+1} given the path of observations $\mathbf{Y}_0^{k+1}$ up to instant $k+1$. This is what our proposed nonlinear stochastic filter computes at each instant.

The first factor can be simplified to

$$P(\mathbf{Z}_0^k \mid z_{k+1}, \mathbf{Y}_0^{k+1}) = P(\mathbf{Z}_0^k \mid z_{k+1}, \mathbf{Y}_0^k). \tag{54}$$

Recall that Barniv's state estimate is given by

$$\hat{z}_{k+1|k+1} = \arg\max_{z_{k+1}} \left\{ \max_{\mathbf{Z}_0^k} P(\mathbf{Z}_0^{k+1} \mid \mathbf{Y}_0^{k+1}) \right\} = \arg\max_{z_{k+1}} \left\{ \max_{\mathbf{Z}_0^k} P(\mathbf{Z}_0^k \mid z_{k+1}, \mathbf{Y}_0^k) P(z_{k+1} \mid \mathbf{Y}_0^{k+1}) \right\}. \tag{55}$$

Note now that

$$\max_{z_{k+1}} \left\{ \max_{\mathbf{Z}_0^k} P(\mathbf{Z}_0^k \mid z_{k+1}, \mathbf{Y}_0^k) P(z_{k+1} \mid \mathbf{Y}_0^{k+1}) \right\} \neq \left[\max_{\mathbf{Z}_0^k} P(\mathbf{Z}_0^k \mid \mathbf{Y}_0^k) \right] \max_{z_{k+1}} P(z_{k+1} \mid \mathbf{Y}_0^{k+1}). \tag{56}$$

If the factorization in (56) were possible, then Barniv's estimate $\hat{z}_{k+1|k+1}$ and ours would coincide. However, because the maximization on the left-hand side of (56) does not factor as the expression on the right-hand side of the same equation, the two estimates may be different. Also note that we provide in our paper only the *filtering* estimate for the unknown state path, i.e., our algorithm computes the sequence

$$\hat{z}_{k|k} \qquad \text{for} \quad k \geq 0.$$

Reference [3] on the other hand provides the *smoothed* state path estimate, i.e., the sequence

$$\hat{z}_{i|k} \qquad \text{for} \quad k \geq 0, \quad i \leq k.$$

The smoothed estimates in [3] are obtained using the backward retrieval step of the Viterbi algorithm, see [10]. In terms of applications, Barniv's algorithm provides a *batch* estimate of the state path,

$$\widehat{\mathbf{Z}_0^k} = \arg\max_{\mathbf{Z}_0^k} P(\mathbf{Z}_0^k \mid \mathbf{Y}_0^k)$$

whereas ours is an *on-line* algorithm that is similar in nature to Kalman–Bucy filtering, i.e., whenever a new state estimate is available at instant k, we do not go back and reestimate the previous states z_i for $i \leq k$.

Finally, in a multitarget scenario where targets are not assumed a priori to be always present, a multitarget detection step must be added to the tracking algorithm. In Barniv's work, the Viterbi forward recursion is run as if only one single target were present and multitarget detection is done simply by thresholding the function $I(z_k)$ at the last stage of the recursion. All states z_k for which $I(z_k)$ exceeds a certain threshold are assumed to be the final state of one possible target. The state trajectories for each detected target are then retrieved by moving backwards along the path of corresponding surviving

nodes in the Viterbi trellis, see [10, 3] for details. Since this procedure leads to a large number of false detections (roughly 40 detections per target [3]), a post-processing clustering step is used to merge nearby estimated trajectories. In our approach, we expand the state space to include dummy "absent target" states and propagate the joint pmf of all target states, including the dummy states. Multitarget detection is then accomplished using a minimum probability of error M-ary Bayes hypotheses test.

2) *Setup of Problem and Modeling Assumptions*: In the sequel, we contrast briefly the state and observation models used in our work with the models introduced in [3]. In our paper, for targets that are present, the corresponding states are the pixel locations at each sensor frame of the target centroids in the discrete centroid grid. In [3], the states are defined as straight line trajectory segments across a group of $G > 1$ sensor frames that define the stages (instants) for the Viterbi forward recursion. The corresponding observation (measurements) model in [3] involves a differential preprocessing of the original sensor images. After preprocessing, it is assumed in [3] that all residual measurement noise is Gaussian and white. In our work, the measurements are the raw sensor frames themselves, with no preprocessing except for a possible removal of the moving local mean (as explained in Section VB). Instead of using a white Gaussian measurement noise assumption, we take full advantage of the real statistics of the background clutter to improve detection/tracking performance. That includes exploring both the clutter spatial (intraframe) correlation and the clutter's possibly non-Gaussian amplitude statistics.

3) *Computational Complexity*: We compare the computational complexity of the Viterbi algorithm to our proposed Bayesian tracker. Let $z_n \in \tilde{\mathcal{L}}$ be the hidden variable, with the number of elements in $\tilde{\mathcal{L}}$ being denoted by L_1. Define the L_1-dimensional vectors $\mathbf{f}_{k|k}$ and $\mathbf{i}_k$, such that $f_{k|k}(j) = P(z_n = j \mid \mathbf{Y}_0^k)$, and $i_k(j) = I(z_k = j)$, $j \in \tilde{\mathcal{L}}$, where $I(z_k)$ is the function defined in (47). Introduce also the $L_1 \times L_1$ transition probability matrix, $\mathbf{P}_T$, such that $P_T(n, j) = P(z_{k+1} = n, \mid z_k = j)$, $(k, j) \in \tilde{\mathcal{L}} \times \tilde{\mathcal{L}}$. The recursion in (51), that corresponds to the Bayes tracker, can be rewritten in matrix notation as

$$\mathbf{f}_{k+1|k+1} = C_{k+1}\mathbf{S}_{k+1} \odot [\mathbf{P}_T \mathbf{f}_{k|k}] \tag{57}$$

where $\odot$ denotes the pointwise multiplication operator and $\mathbf{S}_{k+1}$ is an $L_1 \times 1$ vector such that $S_{k+1}(j) = p(\mathbf{y}_{k+1} \mid z_{k+1} = j)$, $j \in \tilde{\mathcal{L}}$. On the other hand, Viterbi's forward recursion in (48) is written as

$$\mathbf{i}_{k+1} = \mathbf{S}_{k+1} \odot [\max\{\mathbf{P}_T^l \odot \mathbf{i}_k\}]_{1 \le l \le L_1} \tag{58}$$

where $\mathbf{P}_T^l$ is the lth row of the transition matrix $\mathbf{P}_T$, i.e., $P_T^l(j) = P(z_{k+1} = l \mid z_k = j)$, $j \in \tilde{\mathcal{L}}$. The bracketed expression on the right-hand side of (58) reads as follows. For each l, $1 \le l \le L_1$, do the pointwise multiplication of the lth row of the transition probability matrix $\mathbf{P}_T$ by the previous filtering vector $\mathbf{i}_k$ resulting in an L_1-dimensional vector, $\mathbf{i}_k^l$. Then, look for the maximum of the entries $i_k^l(j)$ over the range $1 \le j \le L_1$ and assign this maximum to the lth entry of the bracketed vector.

A comparison between (57) and (58) shows that the two recursions differ basically in the computation of the bracketed vector on the right-hand side. The Bayesian tracker involves the multiplication of an $L_1 \times L_1$ matrix by an $L_1 \times 1$ vector, which requires L_1^2 floating point multiplications and $L_1(L_1 - 1)$ floating point additions. On the other hand, the forward recursion of the Viterbi algorithm requires L_1^2 floating point multiplications and L_1 global maximum searches over an L_1-dimensional vector. Those maximum searches require in turn $L_1(L_1 - 1)$ comparisons. The two algorithms therefore trade arithmetic (addition) computational complexity for logic (comparison) computational complexity. We make two additional remarks.

REMARK 1 The Viterbi smoother requires that, in addition to the forward propagation of $I(z_k)$ using recursion (58), we must also store the indices of the maxima over j, $1 \le j \le L_1$, of $i_k^l(j)$, for all $k > 0$ and all l, $1 \le l \le L_1$. This table of stored indices is necessary for the implementation of Viterbi's backward retrieval step, see [10].

REMARK 2 In most applications, the transition probability matrix $\mathbf{P}_T$ is not a full $L_1 \times L_1$ matrix, as transitions are only alllowed between adjacent target states. As a result of the sparse nature of $\mathbf{P}_T$, the number of floating point multiplications required in the prediction step for both the Bayes tracker and the Viterbi recursion falls in practice from $O(L_1^2)$ to $O(\alpha L_1)$, where $\alpha \ll L_1$. The required number of floating point sums in the prediction step of the Bayes tracker and the complexity of the maxima searches in the Viterbi recursion are also reduced accordingly.

F. Flowchart Summary of the Bayes Detector/Tracker

We present in Table I a flowchart summary of the proposed optimal Bayes detector/tracker.

G. Illustrative Example: Two Extended Targets in Gaussian Noise

Finally, we close this section with an illustrative example of application of the optimal Bayes detector/tracker in a multitarget scenario with overlapping targets. We track/detect two extended targets in a 1D finite grid, against a white Gaussian background clutter with covariance matrix $\sigma_w^2\mathbf{I}$. Both

TABLE I
Flowchart Summary of Multitarget, Multiframe Bayes Detector/Tracker

1) Initialize $P(\mathbf{Z}_n \mid \mathbf{Y}_0^{n-1})$ with the given prior $P(\mathbf{Z}_0)$.
2) Compute $P(\mathbf{Z}_0 \mid \mathbf{Y}_0)$ using equation (26) with $n = 0$.
3) For $n = 1$ up to the total number of available frames:
   - Compute $P(\mathbf{Z}_n \mid \mathbf{Y}_0^{n-1})$ using equation (23).
   - Compute $P(\mathbf{Z}_n \mid \mathbf{Y}_0^{n})$ using equation (26).
   - Do M-ary detection using the hypotheses test (27).
   - If hypothesis H_m is declared true, compute $\Pi_{n|n}^{m}(\mathbf{Z}_n)$ using (35) and look for its maximum over $\mathbf{Z}_n$ to estimate the centroid positions of detected targets.

   End of for-loop.

class 1 and class 2 targets extend over 9 resolution cells with $l_i = r_i = r_s = l_s = 4$, but have different (deterministic) signatures. Class 1 targets have a rectangular-shaped signature, whereas class 2 targets have a triangular-shaped signature.

In any given sensor frame, either two targets (one from class 1, the other from class 2) are present, or just one target (either class 1 or 2) is present, or no target is present. When two targets are present, the corresponding sensor returns may be apart from each other, as shown in Fig. 2. Otherwise, they may overlap in the sensor, causing their signatures to be added in the sensor image, as shown in Fig. 3.

The targets have translational motion with the position of the targets centroids in the 1D grid described by known first-order discrete Markov chains with deterministic drifts $d_1 = 2$ and $d_2 = 3$ for class 1 and class 2 targets, respectively. Once a target belonging to a given class disappears from the sensor range, another target of the same class may appear randomly at any resolution cell with a probability $p_a = 0.3$. The simulation was conducted for a total of 100 frames, with 100 resolution cells per frame. A target that is estimated to be absent is indicated by a "+" mark on the horizontal axis, while a true absence of target is indicated by a "o" mark on the same axis.

Figs. 4(a) and (b) show the tracked positions of the centroids of class 1 and class 2 targets, respectively, between frames 20 and 70, with peak signal-to-noise ratio (PSNR = $10\log_{10}(1/\sigma_w^2)$) equal to 10 dB. Notice that, between frames 65 and 70, class 1 targets are absent from the sensor view, which is correctly indicated by the detector/tracker as a series of superimposed o and + marks on the horizontal axis in Fig. 4(a). During the same time period, a class 2 target is present and is correctly tracked as indicated in the bottom right corner of Fig. 4(b). The opposite situation occurs between frames 36 and 42 when the class 2 target is absent and is correctly declared not present by the detector/tracker (as seen in Fig. 4(b)), but the class 1 target is present and accurately tracked as indicated in Fig. 4(a). Finally, notice that between frames 43 and 54, not only are

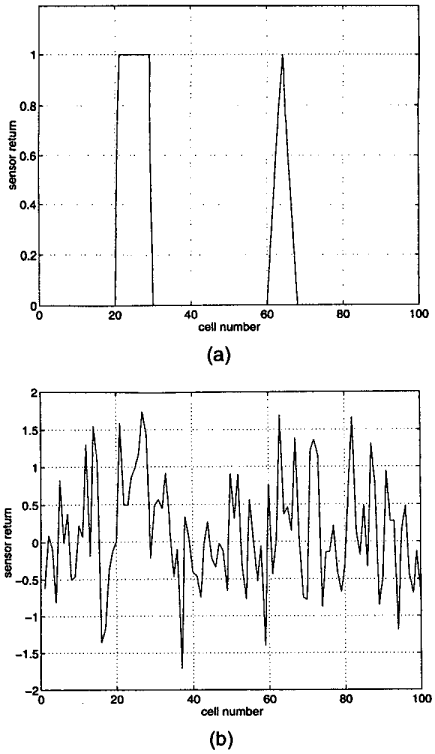


Fig. 2. (a) Noise-free sensor scan with two targets. (b) Observed (noisy) sensor scan, PSNR = 3 dB.

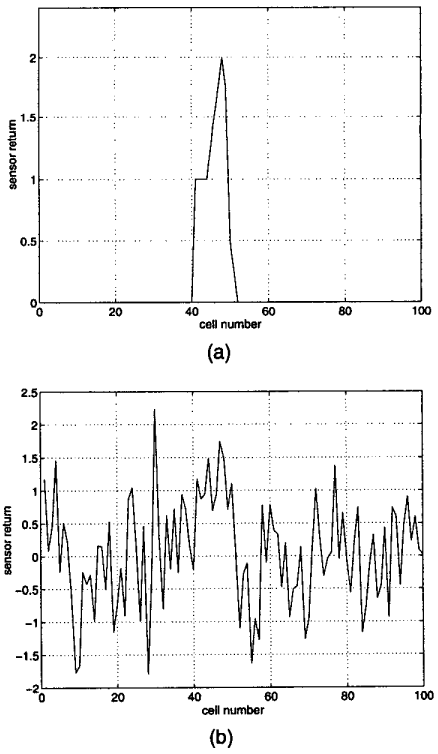


Fig. 3. (a) Noise-free sensor scan with superimposed targets. (b) Observed sensor scan, PSNR = 3 dB.

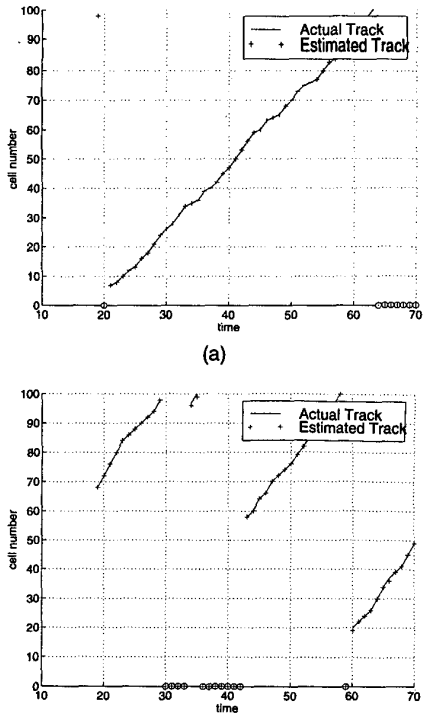


Fig. 4. (a) Centroid tracking for class 1 targets. (b) Centroid tracking for class 2 targets, PSNR = 10 dB.

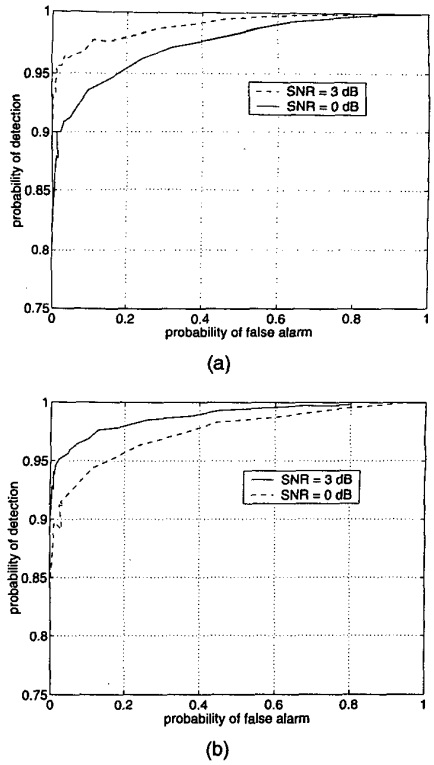


Fig. 5. Performance of multiframe Bayes detector with correlated Gaussian targets in correlated GMRF clutter. (a) $\sigma_{\text{targ}} = 0.2$. (b) $\sigma_{\text{targ}} = 0.4$

both targets present, but also their images overlap since the difference between their centroid positions is less than $2l_s + 1$. Despite the superposition, Fig. 4(a) and (b) show that the centroids of the two targets are accurately tracked independently. The algorithm is capable of performing data association with a high degree of accuracy at the same time that it is able to reject false alarms and prevent misses.

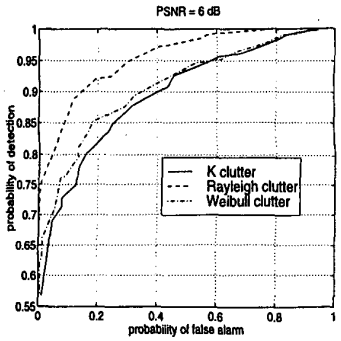
IV. DETECTION PERFORMANCE

We study the detection performance of the optimal nonlinear detector/tracker assuming a single target scenario. The performance is evaluated through the receiver operating characteristic (ROC) curves for the Neyman–Pearson detector, obtained by varying the threshold in (28). The experimental ROCs presented in this section are generated using *Monte Carlo* simulations.

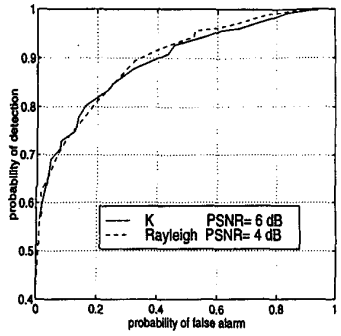
A. Correlated Gaussian Targets in Correlated Gaussian Clutter

We consider first the case when the background clutter is Gaussian and correlated. We simulate 1D Gaussian targets with dimensions $l_i = l_s = 4$ moving in a grid of size $L = 100$. The synthetic targets are samples of a correlated first-order GMRF model with mean $m_a = 1$ and covariance parameters α_a and σ_a. The target image is cluttered by a first order GMRF clutter with parameters α_c and σ_c. The synthetic targets move in a 1D range resolution grid with a mean drift of 2 cells/scan. There is a fluctuation probability of one cell around the mean displacement equal to 0.4. As one target disappears from the surveillance space, there is a 20% probability of a new target reappearing at any arbitrary position in the grid $\mathcal{L} = \{l : 1 \le l \le L\}$, i.e., $p_a = 0.2$. This assumption corresponds to a worst case scenario when new tracks can be initialized with uniform probability at any cell in the sensor grid.

Fig. 5(a) shows the experimental receiving operating characteristic curves (ROCs) for the multiframe Bayes detector, with $\sigma_a = 0.2$, $\alpha_a = 0.16$, and $\alpha_c = 0.24$, for two levels of the average SNR = $20\log_{10}(m_a/\sigma_c)$, respectively 3 and 0 dB (i.e., $\sigma_c = 0.7$ and $\sigma_c = 1$). Fig. 5(b) shows the ROCs when $\sigma_a = 0.4$ (i.e, increasing the variance of the target pixels). The experimental curves were estimated from a total of 8,000 Monte Carlo runs. The plots in Fig. 5 indicate good detection performance, even in the adverse conditions of heavy clutter. For example, the algorithm reaches a 90% probability of detection for false alarm rate of 10^{-3}. Fig. 5(b) shows that there is a slight deterioration in performance when we increase the target variance.



(a)



(b)

Fig. 6. Comparison of optimal Bayes detector performance (a) under Rayleigh, Weibull ($c = 1.5$, $a = 0.6777$), and K ($\nu = 1.5$) clutter, PSNR = 6 dB; (b) under K ($\nu = 1.5$) clutter (PSNR = 6 dB) and Rayleigh clutter (PSNR = 4 dB).

B. Non-Gaussian Clutter

To evaluate the detection performance for non-Gaussian clutter, we ran Monte Carlo simulations with a succession of single targets moving in uncorrelated complex clutter with K and Weibull envelope statistics. For simplicity, the successive targets are pointwise with a unit signature in the in-phase component. We use as figure of merit the PSNR ratio defined as PSNR = $10\log_{10}(1/\sigma_w^2)$, where σ_w^2 is the variance of the white noise sequence from which the corresponding non-Gaussian SIRV is simulated (see reference [16] for details). The motion parameters are the same as in the Gaussian simulation, except that, in the case of complex clutter, a speed of d resolutions cells/scan corresponds to a speed of $2d$ in the double-sized sensor image. The probabilities of detection and false alarm for each value of the threshold are obtained from statistics collected from 10,000 sensor frames, where each frame corresponds to 64 resolution cells or 128 complex quadrature returns.

Fig. 6(a) shows the superposition of the optimal Bayes ROCs with Rayleigh, Weibull, and K clutters, respectively, from top to bottom, for PSNR = 6 dB. The simulated K clutter has shape parameter $\nu = 1.5$. The parameters for the simulated Weibull clutter are $c = 1.5$ and $a = 0.6777$. The plot shows that, even when the correct clutter statistics are incorporated into the detector's structure, there is still a slight deterioration in performance under non-Rayleigh clutter, probably due to the heavier tails of the K and Weibull statistics, which increase the likelihood of false alarms. Fig. 6(b) shows the superposition of the ROC under Rayleigh clutter for PSNR = 4 dB and the ROC under K clutter, PSNR = 6 dB. The two curves in Fig. 6(b) roughly coincide, thus suggesting an approximate 2 dB gain in PSNR under Rayleigh clutter compared with the spiky K clutter.

V. 2D DETECTION/TRACKING PERFORMANCE

We investigate in this section the performance of the proposed Bayes detector/tracker with 2D extended targets moving in digital images corrupted by heavy clutter. We carry out two sets of experiments: one with 2D synthetic data and the other with a real data clutter intensity image recorded by an airbone IR laser radar [17]. In the second set of experiments, a simulated target (military vehicle) template is inserted into the real data background.

A. 2D Simulations with Synthetic Clutter

Clutter: The background clutter is a 2D noncausal first order GMRF. The clutter intensity at pixel (i,j) during the nth scan, $v_n(i,j)$ is modeled by its minimum mean square error (MMSE) representation [29]

$$v_n(i,j) = \beta_h[v_n(i,j-1) + v_n(i,j+1)] + \beta_v[v_n(i-1,j) + v_n(i+1,j)] + u_n(i,j) \tag{59}$$

where $u_n(i,j)$ is the driving noise term. We collect the clutter samples $v_n(i,j)$ and the error samples $u_n(i,j)$ in two long row-lexicographed vectors, respectively $\mathbf{v}_n$ and $\mathbf{u}_n$, thus obtaining an equivalent matrix representation to the difference equation (59). The matrix representation is analogous to the 1D case, i.e.,

$$\mathbf{A}\mathbf{v}_n = \mathbf{u}_n \tag{60}$$

where $\mathbf{A}$ is the potential matrix of the 2D GMRF. Using the orthogonality between the field $\{v_n(i,j)\}$ and the driving noise $\{u_n(i,j)\}$, the clutter covariance matrix is proportional [25] to the inverse of the potential matrix $\mathbf{A}$.

The parametric structure of the potential matrix extends naturally to the 2D case [23, 25]. The corresponding inverse of the clutter covariance for the 2D clutter background is a block-Toeplitz, block-banded matrix where each of the individual blocks is itself Toeplitz and banded [25]. A comprehensive study of the eigenstructure of

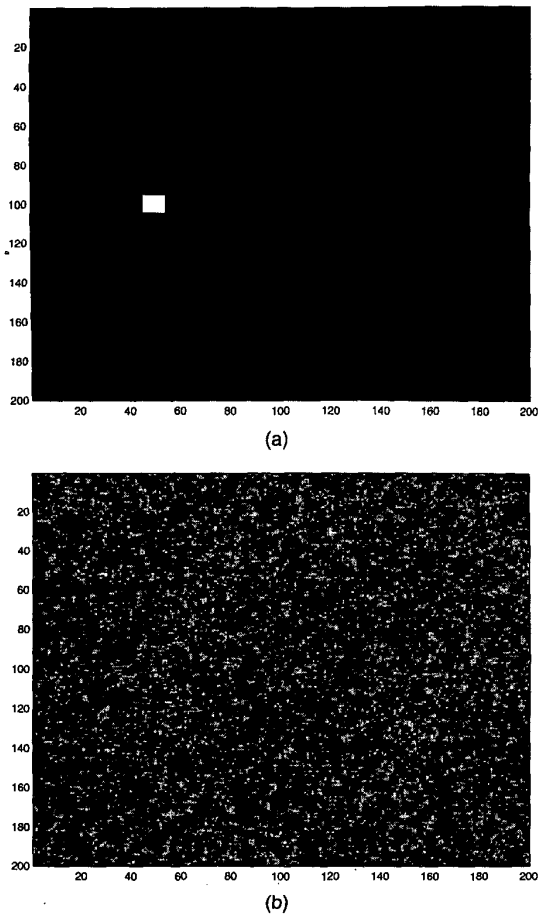


Fig. 7. (a) Clutter-free target image. (b) Simulated sensor image, PSNR = 0 dB.

perturbed Toeplitz and block-Toeplitz matrices and their relation to 1D and 2D GMRF models of *arbitrary order* with different choices of BCs is found in [23].

Target and Observations: For simplicity, we limit our discussion in this section to a single target scenario. The target is a rigid body with a 2D translational motion. We assume that, at any given frame, the target's clutter-free image is contained inside a 2D rectangular region of size $(r_i + r_s + 1) \times (l_i + l_s + 1)$. In this notation, r_i and r_s denote the maximum vertical pixel distances in the target image when we move away, respectively up and down, from the target centroid. Analogously, l_i and l_s denote the maximum horizontal pixel distances in the target image when we move away, respectively left and right, from the target centroid. Let I be the 2D finite lattice $I = \{(k,l): -r_i \le k \le r_s,\ -l_i \le l \le l_s\}$. The 2D target signature at frame n is given by the the signature coefficients

$$a^n(k,l) = c^n(k,l)\phi^n(k,l) \qquad (k,l) \in I \qquad (61)$$

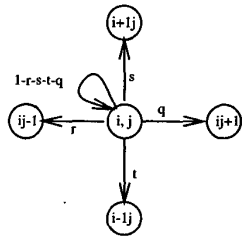


Fig. 8. 2D random walk fluctuation around average target drift.

where the term $\phi^n(k,l) \in \Re$ specifies the target's *pixel intensity* whereas $c^n(k,l) \in \mathcal{B} = \{0,1\}$ is a binary shape coefficient. In the Monte Carlo simulations presented in this section, we assume that the targets have a rectangular template and that their signatures are deterministic, time invariant, and known. Without loss of generality, we make the target pixel intensities constant and equal to 1. The targets are contained in a square region of size 9×9, and are cluttered by a first-order, highly correlated GMRF background with $\beta_h = \beta_v = 0.24$. Figs. 7(a) and (b) show, respectively, the clutter-free target image and a random sample of the target plus clutter image when the target is centered at pixel (100, 50) with PSNR equal to 0 dB.

Motion: We describe here the 2D translational motion model used in the simulations presented in this work. The simulated targets have mean velocity of 2 resolution cells per frame in both the horizontal and vertical directions, and move in a grid of size $L \times L$ pixels. The actual displacement is a random walk fluctuation around the average displacement. In other words, if (i,j) is the predicted centroid position according to the deterministic velocity, the actual position lies in a 2D spatial region around (i,j), with fluctuation probabilities r, s, t, and q shown in Fig. 8. In our simulations, all fluctuation probabilities of one cell were set to 20%. Like in Subsection IIA, we expand the 2D sensor lattice to build a centroid lattice that accounts for boundary effects. We define then an equivalent 1D representation of the 2D centroid lattice that is obtained by sequentially stacking the rows of the 2D lattice into a long vector (see reference [24]). The equivalent 1D centroid lattice is denoted as $\bar{\mathcal{L}}$. Finally, like in the 1D case, we build the augmented lattice $\tilde{\mathcal{L}}$ by adding to $\bar{\mathcal{L}}$ a dummy state that represents the absence of the target. The unknown state at instant n is a 1D random variable z_n defined on $\tilde{\mathcal{L}}$. We denote by $\mathbf{P}_T$ the transition probability matrix that collects the transition probabilities $\{P(z_n = i \mid z_{n-1} = j)\}$, $(i,j) \in \tilde{\mathcal{L}} \times \tilde{\mathcal{L}}$.

Multiframe Bayes Detector/Tracker: Let $\mathbf{y}_n$ be the observed $L \times L$ sensor frame at instant n. Define the filtering posterior probability vector $\mathbf{f}_{n|n}$ such that $f_{n|n}(l) = P(z_n = l \mid \mathbf{Y}_0^n)$, $l \in \tilde{\mathcal{L}}$, where $\tilde{\mathcal{L}}$ now denotes the 1D row lexicographed centroid lattice augmented by

TABLE II
Pseudocode For 2D Bayes Detector/Tracker

a) Initialization.
b) For $n = 1$ to N
- *Prediction step*: $\mathbf{f}_{n|n-1} = \mathbf{P}_T \mathbf{f}_{n-1|n-1}$.
- for $i,j = 1$ to L

  Differential operator: $\mu_{i,j} = y_n(i,j) - \beta_h[y_n(i,j+1) + y_n(i,j-1)] - \beta_v[y_n(i+1,j) + y_n(i-1,j)]$, with $y_n(i,j) = 0$, for $i,j < 1$ or $i,j > L$.

  end of loop
- for $I,J = 1$ to L_1

  $i = I - l_s$; $j = J - l_s$;

  Matched filter: $\lambda_{i,j} = \sum_k \sum_l a_{k,l}\mu_{i+k,j+l}$, with the limits for the summations given in Table III.

  $\mathbf{a}_{i,j} = \{a_{k,l}\}$, with k and l in the ranges assigned to each pair (i,j) in Table III.

  $\mathbf{A}_t = \mathbf{I}_{\bar{r}} \otimes (\mathbf{I}_{\bar{r}} - \beta_h \mathbf{H}_{\bar{l}}) - \beta_v \mathbf{H}_{\bar{r}} \otimes \mathbf{I}_{\bar{l}}$ where $(\bar{r},\bar{l}) = \text{size}(\mathbf{a}_{i,j})$.

  Energy term: $\rho_{i,j} = (\text{vec}[\mathbf{a}_{i,j}])^T \mathbf{A}_t (\text{vec}[\mathbf{a}_{i,j}])$.

  Observations kernel: $S_n((I-1)L_1 + J) = \exp[(2\lambda_{I-l_s,J-l_s} - \rho_{I-l_s,J-l_s})/2\sigma_u^2]$.

  end of loop.
- Normalized kernel entry for absent target state: $S_n(L_1^2 + 1) = 1$.
- *Filtering step*: $\mathbf{f}_{n|n} = C_n \mathbf{S}_n \odot \mathbf{f}_{n|n-1}$ where C_n is a normalization constant such that $\sum_l f_{n|n}(l) = 1$.
- *Binary Detection*: $f_{n|n}(L_1^2 + 1) \underset{H_1}{\overset{H_0}{\gtrless}} 1 - f_{n|n}(L_1^2 + 1)$.
- *MAP estimation*: If hypothesis H_1 (target present) declared true, $\hat{z}_{n|n} = \arg\max_{l \in \tilde{\mathcal{L}}} f_{n|n}(l)$.
- $\mathbf{f}_{n-1|n-1} = \mathbf{f}_{n|n}$.

  End of outer for-loop
c) End of program.

the dummy absent target state. Similarly, introduce the prediction posterior probability vector $\mathbf{f}_{n|n-1}$. Finally, define the observations kernel vector $\mathbf{S}_n$ such that $S_n(l) = p(\mathbf{y}_n \mid z_n = l)$, $l \in \tilde{\mathcal{L}}$. For simplicity of notation, make $r_i = l_i$ and $r_s = l_s$ and introduce $L_1 = L + l_i + l_s$. Let $\mathbf{I}_p$ denote the $p \times p$ identity matrix and $\mathbf{H}_p$ be a $p \times p$ matrix such that $H(i,j) = 1$ for $|i-j| = 1$ or zero otherwise. We use the symbol $\odot$ to denote pointwise multiplication and the symbol $\otimes$ to denote the Kronecker or tensor product [27]. The symbol vec denotes the operator that converts a $P \times Q$ into a PQ-dimensional column vector by sequentially stacking all the rows of the matrix. In the particular case of a single, deterministic target with known and time-invariant signature coefficients $a_{k,l}$, and a first-order 2D GMRF clutter model as described by (60), the optimal 2D Bayes detector/tracker with a total number of frames equal to N, is implemented by the pseudocode in Table II (for a detailed derivation, see [24]).

Remark. The actual implementation of the matrix multiplication $\mathbf{P}_T \mathbf{f}_{n-1|n-1}$ in Table II explores the sparse and block-banded structure of the transition probability matrix $\mathbf{P}_T$. Note also that the energy term ρ is constant for the range $l_i + 1 \le i,\ j \le L - l_s$ and, therefore, can be computed off-line. In general, for an $L \times L$ sensor grid, it can be shown that, using the GMRF clutter model, the Markov chain motion model, and the small extended target models, we reduce total number of required floating point multiplications from $O(L^6)$ to $O(\alpha L^2)$ in the filtering step of the algorithm and, from $O(L^4)$ to $O(\gamma L^2)$ in the prediction step, where $\gamma \ll L$.

We now discuss two suboptimal trackers whose performance we later compare with the performance of our optimal Bayes tracker: the maximum likelihood (ML) tracker, and the linearized Kalman–Bucy tracker.

Maximum Likelihood Tracker: The correlation or memoryless ML tracker ignores the information on the dynamics of the target motion and makes tracking decisions at each sensor scan based solely on the present observed data. In the single target case, let z_n denote as before the position of the centroid of a target that is assumed present in the nth sensor scan. The memoryless ML estimate of the centroid position is given by

$$\hat{z}_{ML} = \arg\max_{i \in \tilde{\mathcal{L}}} p(\mathbf{y}_n \mid z_n = i) \tag{62}$$

where $\tilde{\mathcal{L}}$ is the equivalent 1D centroid lattice.

Linearized Kalman–Bucy Tracker: Although the single frame, memoryless ML tracker is generally accurate in scenarios of high SNR, its performance deteriorates noticeably as the power level of the clutter increases. An improvement to the ML estimator in heavy clutter is obtained by further post-processing its output with a linearized KBF. This is a suboptimal multiframe tracking scheme that reintroduces information about the target dynamics into the tracking algorithm. It resembles common algorithms presented in the literature [1].

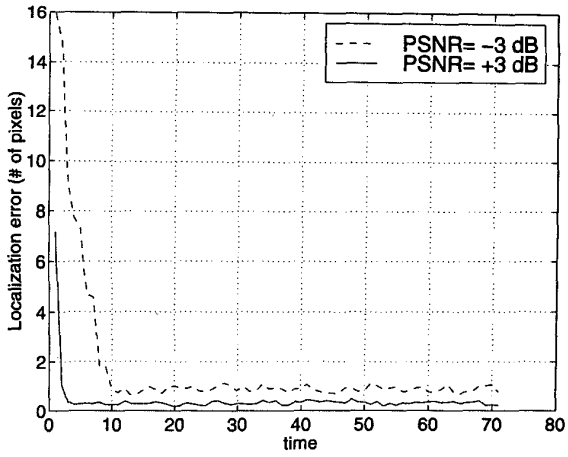
Remark: Computational Complexity. We compare the computational cost of the Bayes detector/tracker to

TABLE III
Computation of Data Term λ_{ij}

$\lambda(i,j)$	$-l_s+1 \le j \le l_i$	$l_i+1 \le j \le L-l_s$	$L-l_s+1 \le j \le L+l_i$
$-l_s+1 \le i \le l_i$	$\sum_{k=-i+1}^{l_s}\sum_{l=-j+1}^{l_s}(.)$	$\sum_{k=-i+1}^{l_s}\sum_{l=-l_i}^{l_s}(.)$	$\sum_{k=-i+1}^{l_s}\sum_{l=-l_i}^{L-j}(.)$
$l_i+1 \le i \le L-l_s$	$\sum_{k=-l_i}^{l_s}\sum_{l=-j+1}^{l_s}(.)$	$\sum_{k=-l_i}^{l_s}\sum_{l=-l_i}^{l_s}(.)$	$\sum_{k=-l_i}^{l_s}\sum_{l=-l_i}^{L-j}(.)$
$L-l_s+1 \le i \le L+l_i$	$\sum_{k=-l_i}^{L-i}\sum_{l=-j+1}^{l_s}(.)$	$\sum_{k=-l_i}^{L-i}\sum_{l=-l_i}^{l_s}(.)$	$\sum_{k=-l_i}^{L-i}\sum_{l=-l_i}^{L-j}(.)$

the association ML-KBF of the ML tracker with the KBF. The ML step in (62), in the case of deterministic targets observed in correlated Gauss–Markov random clutter, reduces to maximizing over all the target centroid positions (i,j) the quantity $\lambda_{i,j}-\rho_{i,j}/2$, where $\lambda_{i,j}$ and $\rho_{i,j}$ are the data and energy terms described in Table II. For an $L\times L$ sensor grid with $L \gg l_i+l_s$, this cost is dominated by the computation of $\lambda_{i,j}$ that is an operation of order $O(\beta L^2)$ with $\beta \ll L$. The computational cost of the KBF stage is neglible in comparison. For a scalar dynamic model in both dimensions, the cost is only 4 floating point multiplications and 4 floating point additions. Hence, the overall ML-KBF association has cost $O(\beta L^2)$. On the other hand, as discussed before, the Bayes detector/tracker requires $O(\alpha L^2)$ floating point multiplications in the filtering step, and $O(\gamma L^2)$ floating point multiplications in the prediction step, with $\alpha \approx \beta \approx \gamma \ll L$. In summary, the Bayes detector/tracker costs roughly twice the cost of the ML-KBF association. The ML-KBF association saves the prediction step of the Bayes detector/tracker.

Tracking Performance in Correlated Clutter: We study first the tracking performance of the Bayes algorithm using synthetic data. The simulated targets are 2D rectangular objects with constant pixel intensity shown in Fig. 7(a). At each sensor scan, we assume that at most one target is present. The target starts from an *unknown* random location in the 50×50 upper left corner of the image and is subsequently tracked over 70 consecutive sensor frames. Fig. 9(a) shows the evolution over time of the standard deviation of the error in the centroid's vertical position estimate given by the nonlinear Bayes tracker. The standard deviation is expressed in number of pixels and evaluated by averaging the errors over 150 Monte Carlo runs. We repeat the experiment for two values of PSNR, respectively +3 dB and −3 dB. The corresponding curves for the horizontal position estimate are qualitatively similar and are omitted for conciseness. Fig. 9(a) shows that there is an initial localization error which declines over time



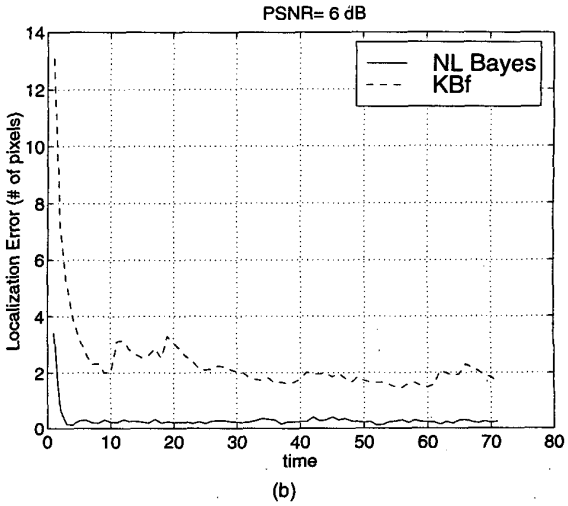


Fig. 9. (a) Performance of nonlinear Bayes tracker in correlated GMRF clutter for PSNR = −3 and +3 dB. (b) Performance of nonlinear Bayes tracker versus linearized KBF, for PSNR = 6 dB.

as new measurements become available. The target acquisition time (i.e., the number of sensor scans for the error to reach its steady state) increases as the PSNR decreases. Likewise, the initial and steady state localization errors also increase with decreasing PSNR.

Next, we compare the nonlinear Bayes tracker with the alternative suboptimal association of the spatial matched filter and a linearized KBF. Fig. 9(b) plots the standard deviation over time of the error in the vertical position estimate for both trackers, with PSNR equal to 6 dB. We see from the plot that the KBF tracker has higher initial and steady state position estimate errors and a longer target acquisition time. Fig. 10 shows again the vertical position estimate error curves for both trackers. This time, the KBF curve is obtained for PSNR equal to 6 dB, while, for the nonlinear Bayes tracker, we lower the PSNR to −5 dB. The two curves in Fig. 10 show that the

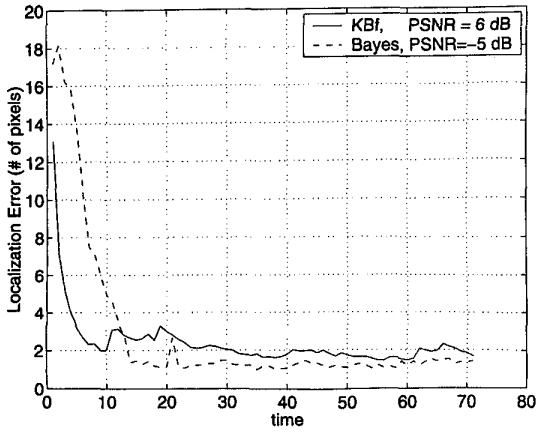


Fig. 10. Comparison of tracking performance of KBF tracker with PSNR = 6 dB and nonlinear Bayes tracker with PSNR = −5 dB.

steady state performance of the nonlinear Bayes tracker and the linearized KBF are very similar, despite an 11 dB difference in PSNR.

B. Detection/Tracking Example with Real Clutter

In order to have a qualitative assessment of the effect of model/data mismatch on the performance of the algorithm, we ran a small-scale simulation with real clutter data. We used real-world intensity imagery of a snow-covered field in Stockbridge, NY, obtained by a 0.85 μm down-looking laser radar [17] mounted to the bottom of a Gulfstream G-1 aircraft. The imagery is from the Infrared Airbone Radar (IRAR) collection at MIT Lincoln Laboratory and was obtained through the website of the Center for Imaging Science at Johns Hopkins University [32]. Fig. 11(a) shows a 120×120 real clutter gray-level intensity image with a heavily cluttered military vehicle (tank) model superimposed on it. Fig. 11(b) shows the tank template alone as a binary image with target intensity equal to 1 and background intensity equal to zero. The tank template (shape) was extracted from a real image of the vehicle taken at the same field with the same sensor. The target pixel intensity was set arbitrarily to achieve the desired low level of contrast between target and clutter. In order to assess tracking performance, we simulated a random trajectory for the target template and detected/tracked it over 27 frames using the Bayes algorithm. The target starts from an unknown location in the 120×120 image and moves in the real clutter background according to a 2D random walk model whose parameters are known to the tracker.

Since the clutter background is real data, we initially preprocess each frame in the image sequence. The preprocessing consists of the segmentation of the original images and the subsequent removal of the spatially-variant local mean in each subimage so that

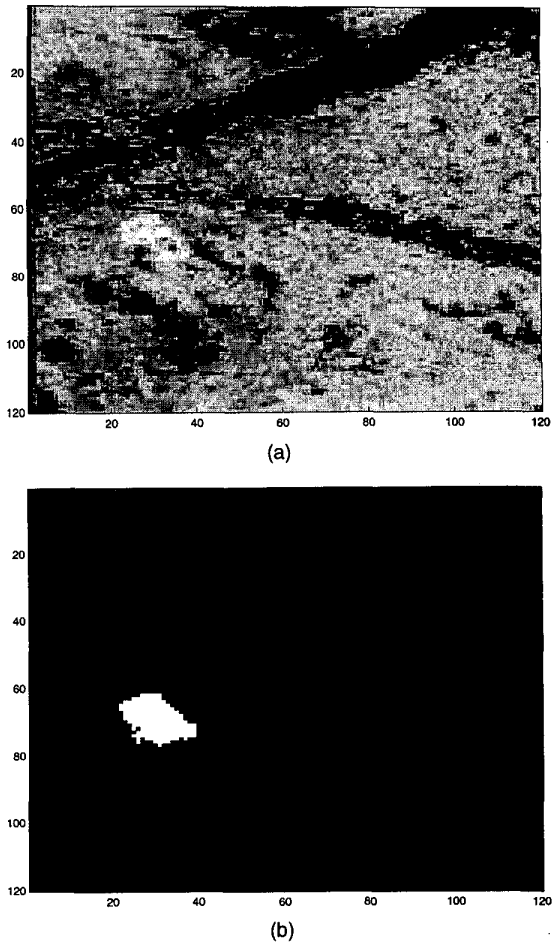


Fig. 11. Simulated real target in heavily cluttered real background (IRAR imagery). (a) Target plus clutter. (b) Target template.

the pixel intensity histogram approaches a zero-mean Gaussian distribution [4]. We then adjust a first-order GMRF model to the Gaussianized data, estimating the corresponding parameters β_h, β_v, and σ_u for each frame. The 2D GMRF parameters were estimated using a simplified version of the ML estimation algorithm introduced in [26].

We compare the tracking results using 1) the proposed Bayes tracker, and 2) a standard 2D image correlator associated to a linearized KBF. The corresponding estimated target trajectories are shown in Fig. 12. The Bayes tracker assumes a uniform initial target position distribution over the entire sensor grid. The linear filter, on the other hand, is initially favored by using a Gaussian initial position prior that is centered in the vicinity of the true initial position and has a small variance. The real simulated trajectory is shown in solid line. The position estimates generated by the Bayes tracker are indicated by the symbol +, whereas the estimates generated by the linearized KBF are interpolated using dashed lines. In the first half of the trajectory shown in Fig. 12, the

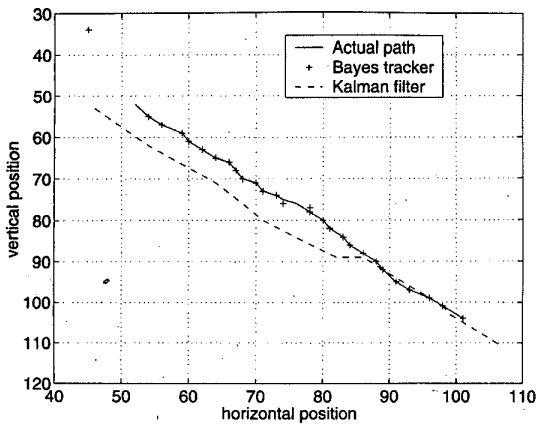


Fig. 12. Nonlinear Bayes detector/tracker versus linearized KBF: performance comparison.

simulated tank is going through a heavily cluttered section of the background, and the single frame standard image correlator is unable to track the target. The KBF tends to discard the correlator's position estimates and through the inertia in its prediction step, tries to fit a straight line trajectory. In the second half of the simulation, when the tank is on an open field, the image correlator is capable of correctly locating the target and the filtering step of the KBF slowly forces the estimated trajectory to approach the true trajectory. By contrast, the Bayes tracker, which has no prior knowledge of the initial position, makes a large initial localization error (the isolated + on the top left corner of Fig. 12), but, afterwards, as new frames become available, the tracker immediately acquires the target and tracks it almost perfectly. A comparison shows that, even in steady state, the localization error for the Bayes tracker is lower than for the KBF, while the acquisition time is much shorter.

Remark. The assumption that the sensor frames are uncorrelated in time is unrealistic in practice. However, the good tracking results with real clutter presented in this section lead us to believe that the Bayes detector/tracker exhibits a high degree of robustness to interframe correlation.

VI. CONCLUSION

In this paper, we presented a new optimal recursive algorithm for integrated, multiframe Bayesian detection and tracking of multiple targets that move randomly in heavily cluttered environments. We considered both extended and pointwise targets with deterministic and random signatures. We developed models for target signature and target motion that take into consideration the finite resolution of the sensors and used these models to build a joint framework for detection and tracking that underlies our proposed Bayesian detector/tracker.

Extensive Monte Carlo simulations in 1D and 2D surveillance spaces determined the performance of the optimal Bayes multiframe detector/tracker under both spatially correlated Gaussian clutter and non-Gaussian clutter with heavy tail (K and Weibull) statistics. The detection and tracking characteristics of the optimal Bayes algorithm are an uppper bound to the performance of other suboptimal algorithms. In particular, when tracking stealthy targets (dim targets in heavy clutter) with the optimal Bayes tracker, the Monte Carlo simulations with 2D targets departing from unknown location show substantial steady state tracking performance gains of up to 11 dB over alternative algorithms found in the literature such as spatial correlators (matched filters) and linearized KBFs. Examples with real clutter data and a known inserted target also show better detection/tracking performance than the association of a conventional image correlator and a Kalman–Bucy tracker. These examples illustrate the robustness of the algorithm to possible mismatches between the data and the underlying models assumed by the algorithm.

REFERENCES

[1] Bar-Shalom, Y., and Li, X. (1995)
Multitarget-Multisensor Tracking: Principles and Techniques.
Storrs, CT: YBS Publishing, 1995.

[2] Bethel, R. E., and Paras, G. J. (1994)
A PDF multitarget tracker.
IEEE Transactions on Aerospace and Electronic Systems, **30**, 2 (Apr. 1994), 386–403.

[3] Barniv, Y. (1985)
Dynamic programming solution for detecting dim moving targets.
IEEE Transactions on Aerospace and Electronic Systems, **21**, 1 (Jan. 1985), 144–156.

[4] Chen, J. Y., and Reed, I. S. (1987)
A detection algorithm for optical targets in clutter.
IEEE Transactions on Aerospace and Electronic Systems, **AES-23**, (Jan. 1987), 46–59.

[5] Pohlig, S. C. (1989)
An algorithm for detection of moving optical targets.
IEEE Transactions on Aerospace an Electronic Systems, **25**, (Jan. 1989), 56–63.

[6] Reed, I. S., Gagliardi, R. M., and Shao, H. M. (1983)
Application of three-dimensional filtering to moving target detection.
IEEE Transactions on Aerospace and Electronic Systems, **AES-19**, (Nov. 1983), 898–905.

[7] Schweitzer, S. M., and Moura, J. M. F. (2000)
Hyperspectral imagery: Clutter adaptation in anomaly detection.
IEEE Transactions on Information Theory, **46**, 5 (Aug. 2000), 1855–1871.

[8] Fortmann, T. E., Bar-Shalom, Y., and Scheffe, M. (1983)
Sonar tracking of multiple targets using joint probabilistic data association.
IEEE Journal of Oceanic Enginnering, **OE-8**, 3 (July 1983), 173–184.

[9] Gelfand, S. B., Fortmann, T. E., and Bar-Shalom, Y. (1996)
Adaptive detection threshold optimization for tracking in clutter.
IEEE Transactions on Aerospace and Electronic Systems. **32**, 2 (Apr. 1996), 514–522.

[10] Forney, G. D., Jr. (1973)
The Viterbi algorithm.
Proceedings of the IEEE, **61**, 3 (Mar. 1973), 268–277.

[11] Larson, R. E., and Peschon, J. (1966)
A dynamic programming approach to trajectory estimation.
IEEE Transactions on Automatic Control, **11**, (July 1966), 537–540.

[12] Bellman, R. (1957)
Dynamic Programming.
Princeton, NJ: Princeton University Press, 1957.

[13] Conte, E., and Longo, M. (1987)
Characterisation of radar clutter as spherically invariant random process.
IEE Proceedings, **134**, Pt.F, 2 (Apr. 1987), 191–197.

[14] Conte, E., Longo, M., and Lops, M. (1991)
Modelling and simulation of non-Rayleigh radar clutter.
IEE Proceedings, **138**, Pt. F, 2 (Apr. 1991), 121–130.

[15] Rangaswamy, M., Weiner, D., and Öztürk, A. (1993)
Non-Gaussian random vector identification using spherically invariant random processes.
IEEE Transactions on Aerospace and Electronic Systems, **29**, 1 (Jan. 1993), 111–123.

[16] Rangaswamy, M., Weiner, D., and Öztürk, A. (1995)
Computer generation of correlated non-Gaussian radar clutter.
IEEE Transactions on Aerospace and Electronic Systems. **31**, 1 (Jan. 1995), 106–116.

[17] Bounds, J. K. (1996)
The infrared airbone radar sensor suite.
RLE technical report 610, Cambridge, MA, Dec. 1996.

[18] Eaves, J. L., and Reedy, E. K. (1987)
Principles of Modern Radar.
New York: Van Nostrand Reinhold, 1987.

[19] Mamic, G., Stitt, N., Iskander, R. (1999)
Coherent detection of radar signals in G-distributed clutter.
In *Proceedings IEEE ICASSP'99*, SPTM2.2, **3**, 1999, 1165–1168.

[20] Sekine, M., Ohtani, S. (1981)
Weibull distributed ground clutter.
IEEE Transactions on Aerospace and Electronic Systems. **AES-17**, (July 1981), 596–598.

[21] Jao, J. K. (1984)
Amplitude distribution of composite terrain radar clutter and the K-distribution.
IEEE Transactions on Antennas and Propagation, **AP-32**, (Oct. 1984), 1049–1062.

[22] Oliver, C. J. (1988)
Representation of radar sea clutter.
IEE Proceedings, **135**, Pt. F, 6 (Dec. 1988), 497–500.

[23] Moura, J. M. F., and Bruno, M. G. S. (1998)
DCT/DST and Gauss–Markov fields: Conditions for equivalence.
IEEE Transactions on Signal Processing, **46**, 9 (Sept. 1998), 2571–2574.

[24] Bruno, M. G. S., and Moura, J. M. F. (1999)
The optimal 2D multiframe detector/tracker.
AEÜ International Journal of Electronics and Communications, **53**, 6, Special Issue on Statistical Signal Processing (Dec. 1999), 346–355.

[25] Moura, J. M. F., and Balram, N. (1992)
Recursive structure of noncausal Gauss–Markov random fields.
IEEE Transactions on Information Theory, **IT-38**, 2 (Mar. 1992), 334–354.

[26] Balram, N., and Moura, J. M. F. (1993)
Parameter estimation in noncausal Gauss–Markov fields.
IEEE Transactions on Information Theory, **39**, 4 (July 1993), 1333–1355.

[27] Graham, A. (1981)
Kronecker Products and Matrix Calculus with Applications.
London: Ellis Horwood Limited, 1981.

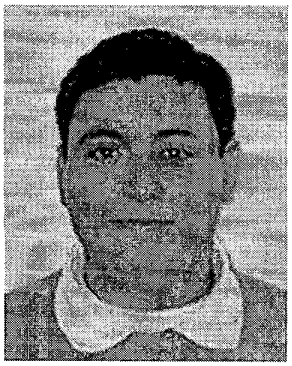
[28] Bucy, R. S., Hecht, C. and Senne, K. D. (1972)
An Enginneer's Guide to Building Nonlinear Filters.
SRL-TR-72-0004, Frank J. Seiler Research Laboratory, May 1972.

[29] Woods, J. W. (1972)
Two-dimensional discrete Markovian fields.
IEEE Transactions on Information Theory, **IT-18**, (Mar. 1972), 232–240.

[30] Jazwinski, A. H. (1970)
Stochastic Processes and Filtering Theory.
New York: Academic Press, 1970.

[31] Van Trees, H. L. (1968)
Detection, Estimation and Modulation Theory, Part I.
New York: John Wiley & Sons, 1968.

[32] Johns Hopkins University,
Center for Imaging Science.
http://www.cis.jhu.edu.



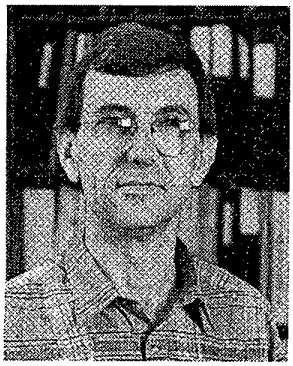
Marcelo G. S. Bruno (M'01) was born in São Paulo, Brazil. He received the B.S. and M.S. degree in electrical engineering from the University of São Paulo, respectively, in 1991 and 1993. He received the Ph.D. degree in electrical and computer engineering from Carnegie Mellon University, Pittsburgh PA, in 1998.

Dr. Bruno is currently with the Communications and Signals Laboratory, Electrical Engineering Department, University of São Paulo, where he is a postdoctoral research fellow. In the summer of 1999, he was a visiting research scholar at Carnegie Mellon University. His research interests are primarily on statistical signal processing, particularly optimal and robust detection, Bayesian and non-Bayesian parameter estimation, array processing, nonlinear stochastic filtering, and stochastic modeling for optimal image and video processing.

José M. F. Moura (S'71—M'75—SM'90—F'94) received the engenheiro electrotécnico degree in 1969 from Instituto Superior Técnico (IST), Lisbon, Portugal, and the M.Sc., E.E., and the D.Sc. in electrical engineering and computer science from the Massachusetts Institute of Technology (M.I.T.), Cambridge, in 1973 and 1975, respectively.

He is a Professor of Electrical and Computer Engineering at Carnegie Mellon University, Pittsubrgh, PA, since 1986. In 1999–2000 he was a Visiting Professor of Electrical Engineering at M.I.T. Prior to this, he was on the faculty of IST (1975–1984), he was Genrad Associate Professor of Electrical Engineering and Computer Science (Visiting) at M.I.T. (1984–1986), and a Visiting Research Scholar at the University of Southern California (Department of Aerospace Engineering, Summers 1978–1981). His research interests include statistical signal processing and telecommunications, image processing, and video representations.

Dr. Moura has published over 230 technical contributions, he is the co-editor of two books, holds four patents on image and video processing, and digital communications, and has given numerous invited seminars at US and European universities and laboratories. He serves as Vice-President for Publications for the IEEE Signal Processing Society, and is a member of the Board of Governors of the same Society. He is also Vice-President for Publications for the IEEE Sensors Council and he is on the Editorial Board of the *IEEE Proceedings*. He Chairs the IEEE Signal Processing Society Publications Board. He was the Editor in Chief for the *IEEE Transactions in Signal Processing* (1975–1999). He is currently a member of the Multimedia Signal Processing technical committee, a founding member of the Sensor Array and Multichannel technical committee, and a past member of the Underwater Accoustics technical committee of the Signal Processing Society. He was on the IEEE Press Board (1991–1995). He Guest Co-Edited a recent issue of the *IEEE Transactions on Information Theory* (August 2000), and has been on the technical program committee of numerous Workshops and Conferences.



He is a Fellow of the IEEE and corresponding member of the Academy of Sciences of Portugal (Section of Sciences). He is affiliated with several IEEE societies, Sigma Xi, AMS, AAAS, IMS, and SIAM. In 2000 he received an IEEE Millenum Medal.