Hyperspectral Imagery: Clutter Adaptation in Anomaly Detection

Susan M. Schweizer, *Member, IEEE,* and José M. F. Moura, *Fellow, IEEE*

***Abstract*—Hyperspectral sensors are passive sensors that simultaneously record images for hundreds of contiguous and narrowly spaced regions of the electromagnetic spectrum. Each image corresponds to the same ground scene, thus creating a cube of images that contain both spatial and spectral information about the objects and backgrounds in the scene. In this paper, we present an adaptive anomaly detector designed assuming that the background clutter in the hyperspectral imagery is a three-dimensional Gauss–Markov random field. This model leads to an efficient and effective algorithm for discriminating man-made objects (the anomalies) in real hyperspectral imagery. The major focus of the paper is on the adaptive stage of the detector, i.e., the estimation of the Gauss–Markov random field parameters. We develop three methods: maximum-likelihood; least squares; and approximate maximum-likelihood. We study these approaches along three directions: estimation error performance, computational cost, and detection performance. In terms of estimation error, we derive the Cramér–Rao bounds and carry out Monte Carlo simulation studies that show that the three estimation procedures have similar performance when the fields are highly correlated, as is often the case with real hyperspectral imagery. The approximate maximum-likelihood method has a clear advantage from the computational point of view. Finally, we test extensively with real hyperspectral imagery the adaptive anomaly detector incorporating either the least squares or the approximate maximum-likelihood estimators. Its performance compares very favorably with that of the RX algorithm, an alternative detector commonly used with multispectral data, while reducing by up to an order of magnitude the associated computational cost.**

***Index Terms*—Anomaly detection, Cramér–Rao bounds, Gauss–Markov random field, hyperspectral imagery, least squares, maximum likelihood, multispectral imagery, ultraspectral imagery.**

I. Introduction

SPECTRAL imaging, also referred to as spectroscopy imaging, exploits multiple regions of the electromagnetic spectrum to probe the composition of a material, or to determine features of interest in a remote scene. In a sense, spectral imaging extends to multiple bands color photographic sensors. Remote sensing uses a variety of sensors flown onboard satellites or aircraft. These sensors employ several bands and have been used in various fields including geology, hydrology, urban planning, geography, cadastral mapping, cartography, and the military. Applications have included the following: remote sensing the earth resources from space, mapping the earth, helping manage water or agricultural resources, monitoring the environment, forestry, detecting and classifying hidden targets in operational theaters. For example, in the U.S., starting in the mid-1960's, with the Earth Resources Survey (ERS) program, and from the early 1970's on with the launching of the first ERS satellite, later renamed Landsat 1 (launched in July 1972), NASA has conceived, designed, and utilized systematically multispectral scanner instruments to sense the earth remotely. Onboard the Landsat satellites were successive generations of multispectral scanner instruments, including the multispectral sensor (MSS) with four bands, the thematic mapper (TM) with seven bands (six in the visible range and one in the thermal range), the enhanced thematic mapper (ETM) with seven spectral bands and one "panchromatic" band (black and white), the enhanced thematic mapper plus (ETM+), and the high-resolution multispectral stereo imager with four high-resolution bands, one panchromatic band, and with stereo capability. Other countries have had their own remote sensing programs including, for example, the "Système Pour L'Observation de la Terre" (SPOT) program, initially proposed by France, now a major international satellite-based remote sensing program—SPOT 1 was launched in February 1986, while SPOT 4 was launched in March 1998. Good book introductions to some of these topics in the context of remote sensing include [1]–[3].

In the above sensors, as well as in several others that were developed in the 80's for both commercial and military purposes, the number of bands is usually relatively small, say below or on the order of ten, and each band is broad. The late 1980's saw the first spectral sensor in remote sensing collecting data in several hundreds of contiguous narrow spectral bands—the airborne visible/infrared imaging spectrometer (AVIRIS)—developed by NASA, tested in 1987, launched in 1989, and currently flown on the NASA ER-2 airplane (a modification of the U2 plane) at an altitude of 20 km, and speed of 730 km/h. Since 1989, several generations of AVIRIS have seen its technology progressively improved. AVIRIS belongs to a new class of sensors, usually referred to as *hyper*spectral sensors to distinguish them from *multi*spectral sensors that employ at most a few tens of spectral bands. Currently, there are a number of hyperspectral sensors, including the hyperspectral digital imagery collection experiment (HYDICE) owned and operated by the U.S. Navy Research Laboratory, and the spatially enhanced broadband array spectrograph system (SEBASS). The latter two are for surveillance missions. There are in addition a number of

Manuscript received July 22, 1999; revised April 12, 2000. The work of S. M. Schweizer was supported by Zonta International through an Amelia Earhart Fellowship Award. The work of J. M. F. Moura was supported in part by ARO under Grant DAAH04-95-1-0494 while on sabbatical at the Laboratory for Information and Decision Systems at the Massachusetts Institute of Technology, Cambridge MA.

The authors are with the Electrical and Computer Engineering Department, Carnegie Mellon University, Pittsburgh, PA 15213-3890 (e-mail: switz@ece.cmu.edu; moura@ece.cmu.edu).

Communicated by J. A. O'Sullivan, Guest Editor.

Publisher Item Identifier S 0018-9448(00)05775-8.

commercial hyperspectral sensors, and also portable hyperspectral cameras.

Anomaly Detector: Hyperspectral sensors collect the spectral signature of a number of contiguous spatial locations (pixels) to form an image—the hyperspectral sensor imagery. The use of this imagery for automatic target detection and recognition is a relatively new area of research. The focus of this paper is on using hyperspectral data for the detection of *anomalous* man-made objects in natural clutter backgrounds. In other words, our goal is to design a detector that flags small regions in the hyperspectral imagery, extending over a few pixels, that correspond to anomalies in a relatively homogeneous background. Hyperspectral imagery shows great potential for this task because it provides both spatial and spectral features about the targets and backgrounds in the imagery. There is reason to believe from published studies, [4]–[6], that the spectral characteristics of natural clutter differ in significant ways from the spectral characteristics of man-made objects, thus hyperspectral sensors can be a tremendous aid in discriminating between the two classes.

One of the challenges of working with hyperspectral imagery is computational: processing efficiently the massive amounts of data collected by the sensor. For example, when sensors are flown on manned or unmanned aircraft it is important to process the data on board to identify regions of interest and reduce the need for transmitting to a ground station a large volume of data. The idea is that only high spatial resolution image chips of these regions of interest are transmitted to the base station.

Many existing algorithms for processing hyperspectral data are the direct outgrowth of algorithms that were developed for single band or for multispectral sensors where the number of bands is small. These approaches resort to a variety of approximations with the goal of simplifying the computational burden. For example, they often ignore the spatial structure of the clutter, working simply with the spectral signature at each pixel of the image. Other methods bin the data, reducing it to a subset of spectral bands. Still others achieve data reduction by some statistical procedure that abstracts from the data a relatively small number of features. Important questions in these approaches include the following: which bands to discard, or, equivalently, the optimal choice of relevant features; which data-reduction statistical procedure to use, e.g., principal component analysis; definition of a good performance metric that can assist in the selection of bands or definition of features.

A widely used anomaly detector for *multi*spectral data, i.e., developed for sensors with a reduced number of spectral bands (ten or less) is the algorithm presented in [7]–[10], commonly referred to as the RX algorithm, after the initials of its proponents, Reed and Xiaoli Yu. The RX algorithm is a likelihood ratio detector based on a number of simplifying assumptions. Extending the RX algorithm to hyperspectral imagery suffers from two major limitations. First, the model assumed by the RX algorithm is limited to clutter that is *spatially* uncorrelated, or spatially white. This model neglects the potentially valuable *spatial* correlation information that is present in the clutter. Second, the RX algorithm does not scale well: it is computationally expensive, since the RX detector requires evaluating the inverse of the sample covariance matrix of the hyperspectral data. This inversion grows with the third power of the dimension of this covariance, i.e., it increases with the cube of the number of spectral bands. Direct application of the RX algorithm to the full assortment of spectral bands in hyperspectral imagery becomes rapidly infeasible. There is a clear need for computationally efficient detectors for hyperspectral sensors that can jointly process all the available spectral bands, and that can exploit simultaneously the spatial and spectral correlation properties of the clutter.

In [11], we present a promising new detector for hyperspectral data—the Gauss–Markov random field (GMRF) algorithm. The GMRF algorithm addresses the concerns described in the above paragraph. First, the GMRF approach models the clutter as a spatially and spectrally colored random field, and so exploits the information on the spatial *and* spectral correlation of the clutter. Secondly, it models directly the inverse of the covariance matrix, rather than the covariance matrix. Finally, recall that the probability density of multivariate Gaussian data, and hence the likelihood function, is expressed in terms of the inverse of the covariance matrix of the data. Since the GMRF detector has an explicit representation of this inverse, it then avoids the inversion of large covariance matrices, which hinders the RX algorithm and other likelihood ratio detectors.

Fully Adaptive Detector—Clutter Statistics: In practice, the clutter is unknown to the anomaly detector. Our emphasis in this paper is in designing a fully *adaptive* GMRF anomaly detector, i.e., an algorithm that adapts to the unknown characteristics of the background (clutter). We consider *generalized* detectors where the unknown data statistics are estimated from the data and used in the detector. To make the detector fully adaptive, we fit the inverse of the covariance matrix of the hyperspectral data to the clutter. This is highly appropriate since, as observed in the previous paragraph, the likelihood functions are expressed directly in terms of this matrix.

With the GMRF assumption, the inverse covariance is highly structured, is sparse, and is described by very few parameters. The main task of the fully adaptive GMRF algorithm is then reduced to estimating this small number of unknown quantities that parameterize the inverse covariance matrix.

The paper studies this estimation problem and its impact on the detector. We present the log-likelihood function, and then note that optimization of this function with respect to the unknown parameters, i.e., maximum-likelihood estimation, is a convex optimization problem—the log-likelihood function is convex on the constrained parameter space, and this parameter space is convex. We establish the Cramér–Rao bounds on the mean-square error of any estimate of the parameters. To compute this bound, we write explicitly the log-likelihood function in terms of the unknown parameters. This leads to an explicit expression for the Fisher information matrix.

The paper then focuses on analyzing the advantages and limitations of methods for approximating the maximum-likelihood (ML) estimates of the GMRF parameters: a direct optimization of the maximum-likelihood function, which will be referred to as ML estimates; a least squares (LS) approach—the LS estimates; and optimization of an approximate log-likelihood function—the approximate ML estimates, or AML estimates. The latter are constructed assuming that the Markov parameters are

close to the boundary of the parameter space. We present the three methods and study their performance in two contexts: as accurate and reliable estimators; and, more importantly, as part of the fully adaptive GMRF anomaly detector. The appropriateness of the estimation techniques for use in the GMRF detector is analyzed in terms of both their computational cost and their detection performance on real hyperspectral data.

Regarding their performance as estimators, we present results with synthetic data, and compare the mean-square error obtained by Monte Carlo simulations with the Cramér–Rao bound. There is good agreement, with some degradation, as expected, of estimation performance for the approximate maximum-likelihood estimates when the parameter values are away from the boundary.

Because the approximate maximum-likelihood estimates and the least squares estimates have similar performance on highly correlated data, and maximum-likelihood estimation is computationally significantly more costly than the other two, when making the GMRF detector fully adaptive, we consider only the least squares and the approximate maximum-likelihood estimation procedures. Our overall conclusion, based on detection results of the fully adaptive GMRF anomaly detector with real hyperspectral imagery, is that the least squares and approximate maximum-likelihood approaches display nearly identical detection performance. Most significantly, the level of performance achieved by the GMRF algorithms even with a reduced number of spectral bands is as good as (if not better than) the performance provided by the RX algorithm, while being appreciably less challenging computationally. In terms of overall detection and computational performance, the results highlight the approximate maximum-likelihood algorithm as a good estimation technique to incorporate in the GMRF anomaly detector for hyperspectral imagery. Because of the good computational characteristics of the GMRF adaptive detector and its good detection performance with real hyperspectral imagery, the GMRF anomaly detector is one of the candidate anomaly detection algorithms to be flown with a new sensor under development by the Adaptive Spectral Reconnaissance Program (ASRP) of the U.S. Defense Advanced Research Projects Agency (DARPA).

The organization of the paper is as follows. In Section II, we present a brief overview of our GMRF modeling framework for hyperspectral clutter. Section III describes the optimal maximum-likelihood approach to estimation. It presents the log-likelihood function, and then uses the GMRF model to derive explicit expressions for this function and to determine the constrained space of the Markov parameters. In Section IV, we derive the least squares method which is a nonparametric estimation procedure, and, in Section V, we detail the approximate maximum-likelihood technique. In Section VI, we analyze the performance of the three estimation methods. We compute the Cramér–Rao bounds for the unknown, deterministic Markov parameters, and then compute by Monte Carlo simulations the bias and the mean-square error associated with the estimates provided by the three methods. In this section, we also study the computational cost of the anomaly detector when it incorporates each of the three estimation procedures. The detector that we will use is derived in detail in [11], and briefly described in Section VII. The target data are assumed to be stochastic, and are essentially modeled by a three–dimensional (3-D) GMRF for which the Markov parameters are distinguishable from those of the clutter background. We present detection performance results for the adaptive anomaly detector using the alternative estimation methods when processing real hyperspectral imagery from the HYDICE and SEBASS sensors. These experimental results compare very favorably with the detection results of the RX algorithm on the same data, while the GMRF algorithm is significantly faster than the RX algorithm. Finally, Section VIII summarizes the paper.

II. GMRF Clutter Model

Hyperspectral sensor imagery is often described as a data cube where the images of a scene at different wavelengths are stacked together in the wavelength dimension. We can consider the hyperspectral cube as a 3–D finite lattice, where each pixel location in the image cube is referenced by the variables i, j, and k, which indicate the spatial location and the particular spectral band in which the pixel lies. Clutter in hyperspectral imagery is usually highly correlated. After a preprocessing step that removes the spatially varying mean of the hyperspectral data, we model this highly correlated spatial and spectral clutter with a noncausal Gauss–Markov random field (GMRF). In addition to capturing the noncausal property of the data, this clutter model accounts for the important Markov nature of the spatially–spectrally correlated background. We refer the reader to the work in [12]–[14]. These references discuss in depth the Markov property and the specification of two-dimensional Markov random fields (MRF). Modeling the statistics of the clutter with mean removed as a Gaussian field is clearly an approximation. This is often assumed, see for example the work in [7] with multispectral data, to simplify the computational cost of the detector. How good is a detector designed on the basis of such an assumption, as with any other model assumed, it remains, of course, to be judged in the final analysis by testing the detector with real hyperspectral imagery, as done in Section VII. Distributions with heavier tails may be more appropriate and will be considered in future research.

As first suggested by Hunt and Cannon in [15], we assume that the Gaussian process describing the dominant image background has a slowly varying covariance structure. Consequently, processing is done on subblocks of data for which the clutter is assumed to be statistically stationary. The processing region is further divided into sublattices of size $N_i \times N_j \times N_k$ where N_k is equal to the total number of available spectral bands, see Fig. 1. We refer to these sublattices as Markov windows. This is similar to the sectioning done in Besag's coding method, which he discusses in [12]. Basically, we adjust Besag's coding method so that our Markov windows do not overlap. The assumed Markov nature of the data implies that pixels within the same Markov window are all that is necessary to completely describe the intensity of the center pixel of that window, i.e., the Markov windows are mutually independent. The pixels in each Markov window are lexicographically ordered to form a set of *pseudo* independent data vectors.

Within each Markov window, we let x_{ijk}, $1 \leq i \leq N_i$, $1 \leq j \leq N_j$, $1 \leq k \leq N_k$ represent a three-dimensional finite

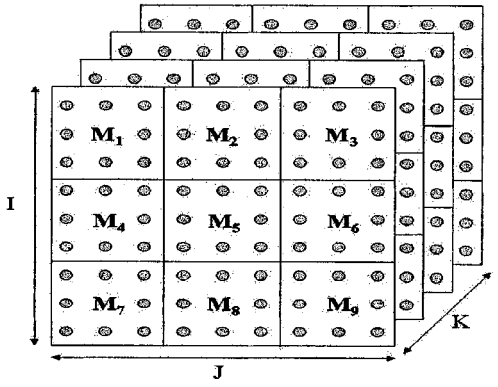


Fig. 1. The hyperspectral image cube is processed on $I \times J \times K$ subcubes in which the clutter is homogeneous. The image is further partitioned into mutually independent Markov windows, denoted by M_p, $1 \le p \le n$ $(n = 9)$.

lattice field. The intensity x_{ijk} of each clutter pixel is described by the minimum mean-square prediction error (MMSE) representation, [16], [17]. This autoregressive formulation is equivalent to the Gibbs distribution formulation, [12]–[14]. For simplicity of presentation, we adopt a first-order, homogeneous, noncausal GMRF described by

$$x_{ijk} = \beta_h(x_{i(j-1)k} + x_{i(j+1)k}) + \beta_v(x_{(i-1)jk} + x_{(i+1)jk}) + \beta_s(x_{ij(k-1)} + x_{ij(k+1)}) + \epsilon_{ijk}. \quad (1)$$

The parameters β_h, β_v, and β_s are the minimum mean-square error predictor coefficients for the spatial and spectral dimensions, respectively, and ϵ_{ijk} is the prediction error. At the boundaries of the Markov window, the model in (1) is completed by appropriate boundary conditions. We assume zero Dirichlet boundary conditions. Other boundary conditions could be adopted, e.g., Neumann or cyclic boundary conditions, see [12], [18]. The noise field, ϵ_{ijk}, is correlated, with correlation structure discussed below. Equation (1) corresponds to a first-order three-dimensional Markov model. It is assumed, without loss of generality, that the clutter is zero mean. In practice, the spatially varying mean is locally estimated and removed from the data. The window size to remove the spatially varying mean is optimized to minimize the third moment of the data, which tends to make the data histogram look closer to a Gaussian density.

Using (1), the data within a Markov window can be compactly represented by the matrix–vector equation

$$A\underline{X} = \underline{\epsilon} \quad (2)$$

where

$$A = \begin{bmatrix} A_1 & A_2 & & & \\ A_2 & \ddots & \ddots & & \underline{0} \\ & \ddots & & \ddots & \\ \underline{0} & & \ddots & \ddots & A_2 \\ & & & A_2 & A_1 \end{bmatrix}. \quad (3)$$

We use the Kronecker product [19], [20] to represent this matrix in a concise manner

$$A = I_{N_k} \otimes A_1 + H^1_{N_k} \otimes A_2 \quad (4)$$

where

$$A_1 = I_{N_i} \otimes B + H^1_{N_i} \otimes C \quad (5)$$

$$A_2 = I_{N_i} \otimes D. \quad (6)$$

The matrices B, C, and D are themselves structured and defined as

$$B = -\beta_h H^1_{N_j} + I_{N_j} \quad (7)$$

$$C = -\beta_v I_{N_j} \quad (8)$$

$$D = -\beta_s I_{N_j}. \quad (9)$$

The symbols I_{N_k}, I_{N_j}, and I_{N_i} are identity matrices, while $H^1_{N_k}$, $H^1_{N_j}$, and $H^1_{N_i}$ are Toeplitz matrices that have zeros everywhere except for the first upper and first lower diagonals, which are composed of all 1's. The subscript denotes the size of the matrices.

The matrix A, referred to as the potential matrix, is a sparse block tridiagonal matrix and contains all the relevant information regarding the GMRF structure [17]. The error vector, $\underline{\epsilon}$, in (2), is a sample from a colored noise process with covariance $\Sigma_\epsilon = \sigma^2 A$. This leads to the direct parameterization of the inverse of the clutter covariance matrix Σ_x^{-1}. Beginning with (2), it can be shown [17] that, by application of the orthogonality principle

$$\Sigma_x^{-1} = \frac{1}{\sigma^2} A. \quad (10)$$

Using (4)–(9) in (10), the inverse covariance matrix of the field $\underline{X}$ is expressed in Kronecker notation as

$$\Sigma_x^{-1} = -\frac{\beta_h}{\sigma^2}\underbrace{(I_{N_k} \otimes I_{N_i} \otimes H^1_{N_j})}_{T_1} + \frac{1}{\sigma^2}\underbrace{(I_{N_k} \otimes I_{N_i} \otimes I_{N_j})}_{T_4} - \frac{\beta_v}{\sigma^2}\underbrace{(I_{N_k} \otimes H^1_{N_i} \otimes I_{N_j})}_{T_2} - \frac{\beta_s}{\sigma^2}\underbrace{(H^1_{N_k} \otimes I_{N_i} \otimes I_{N_j})}_{T_3}. \quad (11)$$

We identify the four main components of the equation by the variables T_1, T_2, T_3, and T_4.

The parameterization of A and Σ_x is a function of σ^2 and of the three Markov parameters, β_h, β_v, and β_s. With higher order Markov models, the parameterization of Σ_x^{-1} would involve a few additional parameters. To make the detector adaptive to the unknown clutter as when using real data, these parameters are estimated from the data. We now turn our focus to the issue of estimation.

III. Maximum Likelihood

A. Parameter Space and Eigenstructure

We first discuss the maximum-likelihood method for estimating nonrandom parameters [21]. Under the GMRF model discussed in Section II, the joint probability density function (pdf) for a set of n independent real-valued data vectors is a multivariate Gaussian density. Using (10), the joint pdf of the independent field samples $\underline{X}_1, \ldots, \underline{X}_n$ factors as

$$p(\underline{X}_1, \ldots, \underline{X}_n|\underline{\theta}) = \frac{|A|^{\frac{n}{2}}}{(2\pi\sigma^2)^{\frac{nN_iN_jN_k}{2}}} \cdot \exp\left(-\frac{1}{2\sigma^2}\sum_{m=1}^{n} \underline{X}'_m A \underline{X}_m\right) \quad (12)$$

TABLE I
PROPERTIES OF LINEAR ALGEBRA AND THE KRONECKER PRODUCT

1.	If $C = A \otimes B$ and the eigenvectors of C, A, and B are ψ_k, x_i, and y_j, respectively, then $\psi_k = x_i \otimes y_j$ where $1 \le i \le m$, $1 \le j \le n$, and $1 \le k \le mn$
2.	If $C = A \otimes B$ and the eigenvalues of C, A, and B are γ_k, λ_i, and μ_j, respectively, then $\gamma_k = \lambda_i \mu_j$ where $1 \le i \le m$, $1 \le j \le n$, and $1 \le k \le mn$
3.	$\rho(A \otimes B) = (\rho A) \otimes B$, where ρ is a scalar
4.	$\text{eig}(\rho A) = \rho(\text{eig}(A))$, where ρ is a scalar and $\text{eig}(A)$ refers to the eigenvalues of A.
5.	$\text{eig}(A + B) = \text{eig}(A) + \text{eig}(B)$ *iff* the eigenvectors of A and B are equal

where $\underline{\mathrm{X}}_m$ is an $N_i N_j N_k \times 1$ observation vector formed from the mth Markov window, q is a 3×1 vector containing the Markov parameters, and $'$ indicates a matrix transpose.

Parameter Space: We start by determining the valid parameter space of the Markov coefficients by analyzing the eigenstructure of the covariance matrix, which is basically the same as analyzing the eigenstructure of its inverse, or of the potential matrix A.

Since A is the inverse of a covariance matrix, it must be a positive-semidefinite matrix. This constraint is equivalent to requiring that the minimum eigenvalue of A be nonnegative. We will look for nondegenerate models and work with A being positive-definite. We determine, analytically, the eigenvalues of the potential matrix.

Beginning with (11), we use properties of linear algebra and the Kronecker product to derive expressions for the eigenvectors and eigenvalues of the potential matrix [19], [22]. Table I lists the properties which we take advantage of in the derivation that follows. We first determine the eigenvectors of the four terms highlighted in (11). We represent the eigenvectors of the matrices $H^1_{N_k}$, $H^1_{N_i}$, and $H^1_{N_j}$ by ϕ_{H_k}, ϕ_{H_i}, and ϕ_{H_j}, respectively. Recalling that any nonzero vector is an eigenvector of the identity matrix [22], we let $\phi_{I_k} = \phi_{H_k}$, $\phi_{I_i} = \phi_{H_i}$, and $\phi_{I_j} = \phi_{H_j}$. Then, by Table I, Property 1, we can choose a common set of eigenvectors for each of the terms T_1, T_2, T_3, and T_4 in (11), which are $\phi_{H_k} \otimes \phi_{H_i} \otimes \phi_{H_j}$. Now, we use Properties 2, 3, 4, and 5 in Table I along with the fact that all eigenvalues of the identity matrix are equal to one to obtain the expression for the eigenvalues $\lambda_{ijk}(A)$

$$\lambda_{ijk}(A) = 1 - \beta_h \lambda_j(H^1_{N_j}) - \beta_v \lambda_i(H^1_{N_i}) - \beta_s \lambda_k(H^1_{N_k}) \quad (13)$$

where $\lambda_j(H^1_{N_j})$, $1 \le j \le N_j$ are the eigenvalues of $H^1_{N_j}$, and similarly for $\lambda_i(H^1_{N_i})$ and $\lambda_k(H^1_{N_k})$.

The matrix H^1_N can be associated with the Even Sine Transform, and has the following set of eigenvalues [23]–[26]:

$$\lambda_p(H^1_N) = 2\cos\left(\frac{p\pi}{N+1}\right), \qquad \text{for } 1 \le p \le N. \quad (14)$$

Combining (14) with (13), the $N_i N_j N_k$ distinct eigenvalues for the potential matrix are completely defined by the Markov parameters β_h, β_v, β_s, and σ^2, and by the dimensions of the cube of data being processed. They are given by

$$\lambda_{ijk}(A) = 1 - 2\beta_h \cos\left(\frac{j\pi}{N_j+1}\right) - 2\beta_v \cdot \cos\left(\frac{i\pi}{N_i+1}\right) - 2\beta_s \cos\left(\frac{k\pi}{N_k+1}\right) \quad (15)$$

where $1 \le j \le N_j$, $1 \le i \le N_i$, and $1 \le k \le N_k$.

Depending on the signs of the Markov parameters, the minimum eigenvalue occurs when the cosine terms are at either their minimum or maximum values. Requiring the minimum eigenvalue of A to be positive translates into the following parameter space constraint:

$$|\beta_h| \cos\left(\frac{\pi}{N_j+1}\right) + |\beta_v| \cos\left(\frac{\pi}{N_i+1}\right) + |\beta_s| \cos\left(\frac{\pi}{N_k+1}\right) < \frac{1}{2}. \quad (16)$$

Equation (16) defines a diamond shape in three-dimensional space.

B. Nonlinear Optimization

With the parameter space defined by (16), we now proceed in deriving the maximum-likelihood estimates.

Structure of the Negative Log-Likelihood Function: Due to the properties of the logarithm function, maximizing (12) with respect to the unknown parameters is the same as minimizing the negative of the log-likelihood function. From (12), the negative log-likelihood function, ignoring constant terms, is

$$L(\underline{X}_1, \ldots, \underline{X}_n | \underline{\theta}) = \frac{nN_iN_jN_k}{2} \ln \sigma^2 - \frac{n}{2} \ln |A| + \frac{n}{2\sigma^2} \operatorname{tr}(AS) \quad (17)$$

where S is the sample covariance matrix

$$S = \frac{1}{n} \sum_{m=1}^{n} \underline{\mathrm{X}}_m \underline{\mathrm{X}}'_m.$$

The second term in (17), which is dependent only on the potential matrix A, is referred to as the *model term*, $M(\underline{\theta})$, and the third term, which is dependent on both A and the data, is referred to as the *data term*, $D(\underline{X}_1, \ldots, \underline{X}_n | \underline{\theta})$.

Using the Kronecker representation for A, which appears in (11), we can express the data term as

$$D(\underline{X}_1, \ldots, \underline{X}_n | \underline{\theta}) = \frac{n}{2\sigma^2} \operatorname{tr}\{[T_4 - \beta_h T_1 - \beta_v T_2 - \beta_s T_3] S\} \quad (18)$$

where T_1, T_2, and T_3 are defined in (11). Equation (18) is equivalent to

$$D(\underline{X}_1, \ldots, \underline{X}_n | \underline{\theta}) = \frac{1}{2\sigma^2} (S_x - 2\beta_h \chi_h - 2\beta_v \chi_v - 2\beta_s \chi_s) \quad (19)$$

where

$$S_x = \sum_{m=1}^{n} \sum_{i=1}^{N_i} \sum_{j=1}^{N_j} \sum_{k=1}^{N_k} (x^m_{ijk})^2 \quad (20)$$

$$\chi_h = \sum_{m=1}^{n} \sum_{i=1}^{N_i} \sum_{j=1}^{N_j - 1} \sum_{k=1}^{N_k} x^m_{ijk} x^m_{i(j+1)k} \quad (21)$$

$$\chi_v = \sum_{m=1}^{n} \sum_{i=1}^{N_i - 1} \sum_{j=1}^{N_j} \sum_{k=1}^{N_k} x^m_{ijk} x^m_{(i+1)jk} \quad (22)$$

$$\chi_s = \sum_{m=1}^{n} \sum_{i=1}^{N_i} \sum_{j=1}^{N_j} \sum_{k=1}^{N_k - 1} x^m_{ijk} x^m_{ij(k+1)}. \quad (23)$$

The quantity x_{ijk}^m represents the intensity of the pixel at spatial location i, j, spectral band k, and within the mth Markov window.

For the model term, we use the product of the eigenvalues, defined in (15), to obtain a parameterization for the determinant of the potential matrix in terms of β_h, β_v, and β_s

$$|A| = \prod_i \prod_j \prod_k \left[1 - 2\beta_h \cos\left(\frac{j\pi}{N_j+1}\right) - 2\beta_v \cos\left(\frac{i\pi}{N_i+1}\right) - 2\beta_s \cos\left(\frac{k\pi}{N_k+1}\right)\right]. \quad (24)$$

Using (24), the model term is expressed as

$$M(\underline{\theta}) = \frac{n}{2} \sum_{i=1}^{N_i} \sum_{j=1}^{N_j} \sum_{k=1}^{N_k} \ln\left[1 - 2\beta_h \cos\left(\frac{j\pi}{N_j+1}\right) - 2\beta_v \cos\left(\frac{i\pi}{N_i+1}\right) - 2\beta_s \cos\left(\frac{k\pi}{N_k+1}\right)\right]. \quad (25)$$

Optimization of the Log-Likelihood Function: The maximum-likelihood estimation of the covariance Σ_x subject to the constraint $\Sigma_x > 0$ is trivially given by

$$\Sigma_x = S.$$

However, our problem belongs to the category of estimating *structured* covariances where the structure of Σ_x is determined by (11). The estimation of structured covariances has been considered in the statistical and signal processing literature, e.g., [27]–[30]. In our case, the constraints are easily expressed in terms of the inverse of the covariance, which is given as a linear combination of known matrices, see (11) and (19).

From (20)–(23), we can see that the data term in the negative log-likelihood function is linear on the parameters.[1] Our estimation of the unknown parameters is a convex optimization problem: the log-likelihood function is convex on the parameter space determined by $A > 0$ (see condition (16)), and the constraint set defined by (16) is convex. The positive-definite constraint on A is commonly referred to as a linear matrix inequality constraint in the parameters. This optimization problem is an instance of a general optimization problem known in the literature as a max-det problem, [31]. This guarantees the existence of a global extremum and no other local minima in the parameter space.

As a result of the logarithmic component present in the model term, the gradient of $L(\underline{X}_1, \ldots, \underline{X}_n|\underline{\theta})$ cannot be explicitly solved for the Markov parameters. However, the maximum-likelihood estimate for the scaling parameter σ^2 can be solved for and is a linear combination of the maximum-likelihood estimates for the three Markov coefficients, $\widehat{\beta_h}$, $\widehat{\beta_v}$, and $\widehat{\beta_s}$

$$\widehat{\sigma^2}_{ml} = \frac{1}{nN_iN_jN_k} \sum_{m=1}^{n} \underline{\mathrm{X}}_m' \widehat{A_{ml}} \underline{\mathrm{X}}_m = \frac{1}{nN_iN_jN_k} (S_x - 2\widehat{\beta_h}\chi_h - 2\widehat{\beta_v}\chi_v 2\widehat{\beta_s}\chi_s) \quad (26)$$

where S_x, χ_h, χ_v, and χ_s are defined by (20)–(23), respectively.

[1]Strictly speaking, the data term is linear on $1/\sigma^2$ and a suitable normalization of the Markov parameters by this quantity.

To perform the optimization for the Markov parameters, we use the Polak–Ribiere conjugate gradient method as implemented in [32]. This implementation performs a series of one–dimensional (1-D) line minimizations rather than a more complex multidimensional optimization. To incorporate the parameter space constraints into the optimization, we use a modification to the bracketing scheme that was suggested by Balram and Moura [23]. The bracketing scheme uses (16) to ensure that the ML estimates remain within the parameter space.

The nonlinear optimization that is required to obtain the ML estimates is computationally expensive, which is not desirable when trying to develop an overall detection algorithm that is implementable in real time. Therefore, we look to alternative estimation procedures in order to reduce the computational load of our GMRF detector.

IV. Least Squares

A. Least Squares Formulation

One alternative approach to the maximum-likelihood technique is the least squares method. This method is based on minimizing the mean-square modeling error

$$\sum_{m=1}^{n} \underline{\epsilon}_m' \underline{\epsilon}_m = \sum_{m=1}^{n} (A\underline{\mathrm{X}}_m)'(A\underline{\mathrm{X}}_m) \quad (27)$$

where n is the total number of independent data vectors within the processing window, and the subscript m denotes from which particular Markov window the data vector was derived. The least squares method provides computational advantages over the optimal approach in two respects: first, it avoids a nonlinear optimization, and, second, since computing the likelihood function is not a part of the estimation procedure, the least squares approach does not necessitate constraining the parameter space.

By taking advantage of the Kronecker representation for A given in (4)–(9), and rearranging terms, we determine that

$$A\underline{\mathrm{X}}_m = \underline{\mathrm{X}}_m - G_m\underline{\theta} \quad (28)$$

where

$$G_m = [T_1\underline{\mathrm{X}}_m \quad T_2\underline{\mathrm{X}}_m \quad T_3\underline{\mathrm{X}}_m] \quad (29)$$

$$\underline{\theta} = [\beta_h \quad \beta_v \quad \beta_s]' \quad (30)$$

and T_1, T_2, and T_3 are the Kronecker expressions defined in (11). Using the above notation, the least squares estimates of the Markov coefficients are [33], [34]

$$\widehat{\theta_{LS}} = \left(\sum_{m=1}^{n} G_m' G_m\right)^{-1} \left(\sum_{m=1}^{n} G_m' \underline{\mathrm{X}}_m\right). \quad (31)$$

The estimate for the scaling parameter σ^2 is obtained by using (26), and replacing $\widehat{A_{ml}}$ with $\widehat{A_{ls}}$.

Since there are three unknown scalar parameters, we see from (31) that least squares estimation requires taking the inverse of a 3×3 matrix. For hyperspectral data, this is significantly less challenging computationally than computing the inverse of a full data covariance matrix, as for example in [7], [35], and is also computationally better than the maximum-likelihood nonlinear optimization presented in Section III. However, to obtain

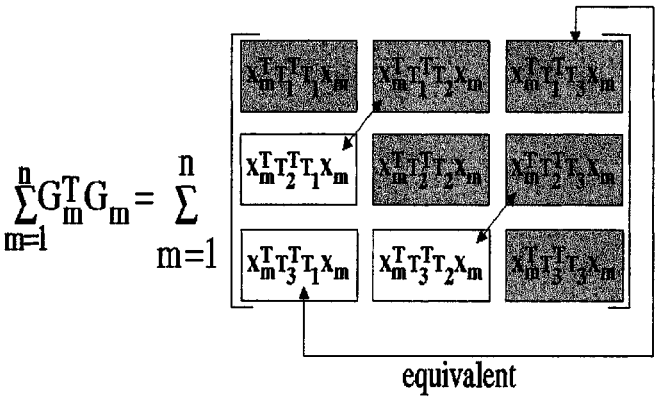


Fig. 2. Unique components of $G'_m G_m$: Only six of the nine components are unique.

the $G'_m G_m$ matrices in (31) through direct vector multiplication still requires significant processing, as well as a large amount of storage. In the next section, we show a means of reducing both the storage and processing required by the least squares approach.

B. Reduced Least Squares

To minimize the processing involved in the least squares estimation method, we take advantage of the fact that the matrix $\sum_{m=1}^{n} G'_m G_m$ is symmetric and so only six of the nine components are unique, see Fig. 2. The matrices T_1, T_2, and T_3 are defined in (11).

We use properties of the Kronecker product [19], [20] to simplify each of the six components into linear combinations of one-step-ahead and two-step-ahead correlations

$$\sum_{m=1}^{n} X'_m T'_1 T_1 X_m = 2S_x - S_{xp} + 2\chi_{h2} \quad (32)$$
$$\sum_{m=1}^{n} X'_m T'_2 T_2 X_m = 2S_x - S_{xp2} + 2\chi_{v2} \quad (33)$$
$$\sum_{m=1}^{n} X'_m T'_3 T_3 X_m = 2S_x - S_{xp3} + 2\chi_{s2} \quad (34)$$
$$\sum_{m=1}^{n} X'_m T'_1 T_2 X_m = 2\chi_{hv} + 2\chi_{hv2} \quad (35)$$
$$\sum_{m=1}^{n} X'_m T'_1 T_3 X_m = 2\chi_{sh} + 2\chi_{sh2} \quad (36)$$
$$\sum_{m=1}^{n} X'_m T'_2 T_3 X_m = 2\chi_{sv} + 2\chi_{sv2} \quad (37)$$

where

$$S_{xp} = \sum_{m=1}^{n} \sum_{i=1}^{N_i} \sum_{k=1}^{N_k} (x^m_{i1k})^2 + (x^m_{iN_jk})^2$$
$$S_{xp2} = \sum_{m=1}^{n} \sum_{j=1}^{N_j} \sum_{k=1}^{N_k} (x^m_{1jk})^2 + (x^m_{N_ijk})^2$$
$$S_{xp3} = \sum_{m=1}^{n} \sum_{i=1}^{N_i} \sum_{j=1}^{N_j} (x^m_{ij1})^2 + (x^m_{ijN_k})^2$$
$$\chi_{h2} = \sum_{m=1}^{n} \sum_{i=1}^{N_i} \sum_{j=1}^{N_j-2} \sum_{k=1}^{N_k} x^m_{ijk} x^m_{i(j+2)k}$$
$$\chi_{v2} = \sum_{m=1}^{n} \sum_{i=1}^{N_i-2} \sum_{j=1}^{N_j} \sum_{k=1}^{N_k} x^m_{ijk} x^m_{(i+2)jk}$$
$$\chi_{s2} = \sum_{m=1}^{n} \sum_{i=1}^{N_i} \sum_{j=1}^{N_j} \sum_{k=1}^{N_k-2} x^m_{ijk} x^m_{ij(k+2)}$$
$$\chi_{hv} = \sum_{m=1}^{n} \sum_{i=1}^{N_i-1} \sum_{j=1}^{N_j-1} \sum_{k=1}^{N_k} x^m_{ijk} x^m_{(i+1)\,(j+1)k}$$
$$\chi_{hv2} = \sum_{m=1}^{n} \sum_{i=1}^{N_i-1} \sum_{j=1}^{N_j-1} \sum_{k=1}^{N_k} x^m_{i(j+1)k} x^m_{(i+1)jk}$$
$$\chi_{sh} = \sum_{m=1}^{n} \sum_{i=1}^{N_i} \sum_{j=1}^{N_j-1} \sum_{k=1}^{N_k-1} x^m_{ijk} x^m_{i(j+1)(k+1)}$$
$$\chi_{sh2} = \sum_{m=1}^{n} \sum_{i=1}^{N_i} \sum_{j=1}^{N_j-1} \sum_{k=1}^{N_k-1} x^m_{i(j+1)k} x^m_{ij(k+1)}$$
$$\chi_{sv} = \sum_{m=1}^{n} \sum_{i=1}^{N_i-1} \sum_{j=1}^{N_j} \sum_{k=1}^{N_k-1} x^m_{ijk} x^m_{(i+1)j(k+1)}$$
$$\chi_{sv2} = \sum_{m=1}^{n} \sum_{i=1}^{N_i-1} \sum_{j=1}^{N_j} \sum_{k=1}^{N_k-1} x^m_{(i+1)jk} x^m_{ij(k+1)},$$

and S_x is defined in (20). Storing the 12 values defined above, rather than the G_m matrices, saves a significant amount of storage, and therefore decreases the overall processing time.

In the next section, in an effort to reduce the computational complexity even further, we look to another estimation procedure, approximate maximum-likelihood (approximate-ML). In addition, the approximate-ML procedure differs from least squares in that it is a parametric approach that takes advantage of the Gaussian pdf for the clutter background.

V. Approximate Maximum Likelihood

For highly correlated three-dimensional fields, we have observed that the maximum-likelihood estimates, as for two-dimensional fields [23], [18], tend to reside on the boundary of the parameter space. In this section, we present an alternative means of estimating the maximum-likelihood parameters that approximates the optimal maximum-likelihood estimates. The approach assumes that real HSI is highly correlated, and takes advantage of the fact that the parameter estimates should be close to the boundary of the parameter space, see also [23]. We refer to this method as the approximate-ML approach.

Approximate-ML makes use of a simple mathematical approximation for the $\ln(1-\eta)$ that is present in the model term defined in (25) in Section III. We use the second-order Taylor approximation for the logarithm

$$\ln(1-\eta) = -\eta - \frac{\eta^2}{2} + \mathrm{o}(\eta^2) \quad (38)$$

where

$$\eta = 2\beta_h \cos\left(\frac{j\pi}{N_j+1}\right) + 2\beta_v \cos\left(\frac{i\pi}{N_i+1}\right) + 2\beta_s \cos\left(\frac{k\pi}{N_k+1}\right). \quad (39)$$

Use of the Taylor series approximation in conjunction with the following two trigonometric properties:

$$\sum_{i=1}^{N} \cos\left(\frac{i\pi}{N+1}\right) = 0$$
$$\sum_{i=1}^{N} \cos^2\left(\frac{i\pi}{N+1}\right) = \frac{N-1}{2}$$

leads to a simplified form for the model term

$$M(\underline{\theta}) = -\sum_{i=1}^{N_i} \sum_{j=1}^{N_j} \sum_{k=1}^{N_k} \left(\beta_h^2 \frac{N_j-1}{2N_j} + \beta_v^2 \frac{N_i-1}{2N_i} + \beta_s^2 \frac{N_k-1}{2N_k}\right). \quad (40)$$

Using (40) in the likelihood function in (17), taking derivatives with respect to the Markov parameters, and equating to zero leads to explicit expressions for $\widehat{\beta_h}$, $\widehat{\beta_v}$, and $\widehat{\beta_s}$

$$\widehat{\beta_h} = \frac{\chi_h}{n\sigma^2 N_i N_k (N_j - 1)} \quad (41)$$
$$\widehat{\beta_v} = \frac{\chi_v}{n\sigma^2 N_j N_k (N_i - 1)} \quad (42)$$
$$\widehat{\beta_s} = \frac{\chi_s}{n\sigma^2 N_i N_j (N_k - 1)} \quad (43)$$

where χ_h, χ_v, and χ_s are referred to as the one-step-ahead correlations, and are defined in (21)–(23), respectively. The above expressions for $\widehat{\beta_h}$, $\widehat{\beta_v}$, and $\widehat{\beta_s}$ are only valid when η is much smaller than one, or, in other words, when the data are *weakly* correlated and the Markov parameters are far from the parameter space boundaries. However, real hyperspectral imagery is known to be *highly* correlated, especially in the spectral dimension. Thus we must adjust expressions (41)–(43) to accommodate for η not being small.

First, from (41)–(43) we obtain expressions for the ratios $\frac{\beta_h}{\beta_v}$ and $\frac{\beta_h}{\beta_s}$. This eliminates the dependency on σ^2, but only provides us with two equations for determining the three unknown parameters β_h, β_v, and β_s. Since with highly correlated data, the parameters should be near to the boundaries of the parameter space, we use the parameter space constraint defined in (16) as our third equation, thus forcing the parameter estimates to be closer to the boundaries. We now have three equations and three unknown parameters. By solving the set of linear equations, we obtain the approximate-ML estimates (see (44)–(46) at the bottom of this page), where

$$\alpha = \frac{N_k(N_j-1)}{N_j(N_k-1)}$$

$\xi = 0.5 - \delta$, and δ is a small number included to ensure that the estimates are inside the parameter space. The estimate for the scaling parameter σ^2 is obtained by using (26), and replacing $\widehat{A_{ml}}$ with $\widehat{A_{aml}}$.

VI. Estimation Performance

In this section, we first compute in Section VI-A the Cramér–Rao bounds for the Markov parameters, when the parameter σ^2 is assumed known. These bounds [21] represent a lower bound on the accuracy of any unbiased estimate of the desired parameters. In Section VI-B, we study the quality of the three methods presented in Section III by computing by Monte Carlo simulations the bias and the mean-square error of the three estimates: maximum-likelihood, least squares, and approximate maximum-likelihood. Finally, in Section VI-C, we compare the three estimation schemes with respect to their computational cost.

A. Cramér–Rao Bounds

We model the three Markov parameters as deterministic, unknown, and collect them in the vector

$$\beta = \begin{bmatrix} \beta_h \\ \beta_v \\ \beta_s \end{bmatrix}.$$

If Γ_ϵ is the error covariance matrix associated with any unbiased estimate of the unknown parameter vector β, the Cramér–Rao bound, see [21], is given by

$$\Gamma_\epsilon \geq J^{-1} \tag{47}$$

where J is the Fisher information matrix

$$J = \left[E\left\{ \frac{\partial^2 L(\underline{X}_1, \ldots, \underline{X}_n | \underline{\beta})}{\partial \beta_i \beta_j} \right\} \right], \qquad i,j = h,v,s \tag{48}$$

where $L(\underline{X}_1, \ldots, \underline{X}_n | \underline{\beta})$ is the negative log-likelihood function given by (17). From (17), we see that the data term in $L(\underline{X}_1, \ldots, \underline{X}_n | \underline{\beta})$ is linear in the Markov parameters and the first term does not depend on them. Since (48) involves second-order derivatives with respect to the parameters, we are only concerned with the middle term in (17), i.e., the model term $(n/2)\ln|A|$. To compute the derivatives of this term, we need to express the determinant of A in terms of the Markov parameters β. This is provided by (25). Using this expression, and taking the second-order derivatives in (48), the Fisher information matrix is

$$J = 2n \sum_{i,j,k} \frac{1}{D^2} \cdot \begin{bmatrix} \cos^2 \frac{j\pi}{N_j+1} & \cos\frac{j\pi}{N_j+1}\cos\frac{i\pi}{N_i+1} & \cos\frac{j\pi}{N_j+1}\cos\frac{k\pi}{N_k+1} \\ = & \cos^2\frac{i\pi}{N_i+1} & \cos\frac{i\pi}{N_i+1}\cos\frac{k\pi}{N_k+1} \\ = & = & \cos^2\frac{k\pi}{N_k+1} \end{bmatrix} \tag{49}$$

where

$$D = \left[1 - 2\beta_h \cos\left(\frac{j\pi}{N_j+1}\right) - 2\beta_v \cos\left(\frac{i\pi}{N_i+1}\right) - 2\beta_s \cos\left(\frac{k\pi}{N_k+1}\right) \right].$$

Inverting the Fisher information matrix given by (49) and replacing this inverse in (47), we get a lower bound on the error covariance Γ_ϵ. For example, the variances of the error estimates are bounded below by

$$\sigma^2_{\beta_i} \geq (J^{-1})_{ii}, \qquad i = h,v,s. \tag{50}$$

These bounds are computed numerically in the next subsection.

B. Estimation Accuracy

We now consider the negative log-likelihood function and then analyze the error accuracy and reliability of the maximum-likelihood, approximate maximum-likelihood, and least squares estimates. To carry out these studies, we generate samples of a three-dimensional noncausal GMRF field for fixed values of β_v, β_s, and σ^2. Each sample is of spatial dimensions 15×15, and is

$$\widehat{\beta_h} = \frac{\xi\chi_h}{|\chi_h|\cos\left(\frac{\pi}{(N_j+1)}\right) + |\chi_v|\cos\left(\frac{\pi}{(N_i+1)}\right) + \alpha|\chi_s|\cos\left(\frac{\pi}{(N_k+1)}\right)} \tag{44}$$

$$\widehat{\beta_v} = \frac{\xi\chi_v}{|\chi_h|\cos\left(\frac{\pi}{(N_j+1)}\right) + |\chi_v|\cos\left(\frac{\pi}{(N_i+1)}\right) + \alpha|\chi_s|\cos\left(\frac{\pi}{(N_k+1)}\right)} \tag{45}$$

$$\widehat{\beta_s} = \frac{\alpha\xi\chi_s}{|\chi_h|\cos\left(\frac{\pi}{(N_j+1)}\right) + |\chi_v|\cos\left(\frac{\pi}{(N_i+1)}\right) + \alpha|\chi_s|\cos\left(\frac{\pi}{(N_k+1)}\right)} \tag{46}$$

comprised of 15 spectral bands. The chosen spatial dimensions match those of the processing window used on real hyperspectral sensor imagery with pixels of 1-m spatial resolution. In all three examples, the scaling parameter σ^2 is set equal to 1. We generate the samples of the noncausal Gauss–Markov random field through a simple extension of a method developed in [17], [18]. In general, the synthesis method uses the Cholesky decomposition of the potential matrix to derive an equivalent one-sided regressor for the noncausal field. We drive this one-sided regressor with Gaussian white noise input, generating recursively a sample of the noncausal three-dimensional field with given parameter values.

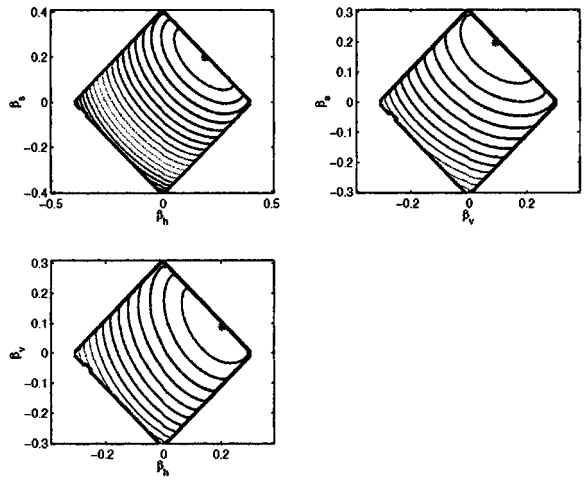
Plots of the Negative Log-Likelihood Function: We illustrate this function in two cases: when the field is highly correlated with Markov parameters $\beta_h = 0.2$, $\beta_v = 0.1$, and $\beta_s = 0.2$; and when the field has a mixed correlation $\beta_h = 0.02$, $\beta_v = 0.01$, and $\beta_s = 0.4$. With the high correlated field, the values of the Markov parameters are such that the constraint in (16) is met with equality. In Fig. 3(a), we show contour plots of the negative log-likelihood function, see (17) on the valid parameter space. Since there are three parameters of interest, the contour plots are generated by fixing both σ^2 and one of the β parameters at their true values, while varying the other two parameters. The minimum point is marked with an asterisk. In the high-correlation example, since the parameter space constraint is met with equality, the true parameter values sit on the boundary of the parameter space.

In Fig. 3(b), we display the contour plots of the negative log-likelihood function for the mixed correlated field. In the spectral dimension, the true parameter value sits close to the boundary of the parameter space, while the true values for the spatial parameters are centrally located in the parameter space. It can be observed in the lower left contour plot in Fig. 3(b), that, for a fixed value of β_s, the log-likelihood function is rather flat, indicating that it is basically insensitive to changes in β_h and β_v.

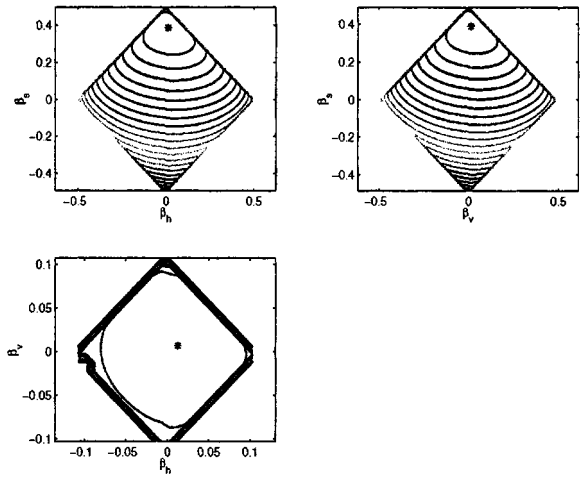
The flat structure of the log-likelihood function in these examples has implications in the mean-square behavior of the estimates.

Mean-Square Error and Cramér–Rao Bound Studies: We now compute by Monte Carlo simulation the experimental bias and mean-square error for each of the three estimates of the Markov parameters: maximum-likelihood (ML) estimate; least squares (LS) estimate; and approximate maximum-likelihood (AML) estimate. In the simulation studies below, we fix the values of two of the Markov parameters $\beta_v = \beta_s = 0.1$, and the value of the scaling parameter $\sigma^2 = 1$. The scaling parameter is assumed to be known. We let β_h take six different values in the range $= [0.05\,0.3]$ at increments of 0.05. Note that the sum of the β's will go from 0.25, which represents a low correlated field, to 0.5, a highly correlated field. Because of the symmetry of the Cramér–Rao bound expression on the β's, we expect to obtain similar results when we repeat these experiments by varying in a similar fashion one of the other parameters rather than β_h. This was essentially confirmed in our studies, and is not illustrated for lack of space.

Fig. 4 shows the (normalized) bias of the three estimates for the parameter β_h computed with MC = 500 Monte Carlo runs. From the figure, we see that the maximum-likelihood (solid line) and the least squares (dash–dotted line) estimates are either unbiased or exhibit a very small bias, except when the configuration of the β parameters is close to the boundary ($\beta_h = 0.3$) (extreme right of the plot) where the maximum-likelihood estimate shows a slight bias. This is expected because we have noticed that the ML function is significantly flat when the Markov parameters are close to the boundary of the parameter space, i.e., the field is highly correlated and the gradient descent algorithm has difficulties in determining the minimum while satisfying the parameter space constraint. On the other hand, the approximate maximum-likelihood estimate (dashed line, $*$ marks) as computed from (44)–(46) improves significantly its performance as we move closer to the boundary, see the bottom plot in the figure that is monotonically improving from left to right. This is intuitively pleasant since, from our discussion in Section V, (44)–(46) were derived assuming a highly correlated field. Given that real images tend to be highly correlated, this observation helps explain why in Section VII the detection results with real hyperspectral imagery using the adaptive GMRF de-



(a)



(b)

Fig. 3. (a) Highly correlated field. The true parameter values sit on the boundary of the parameter space. (b) Mixed correlation field. True β_s value sits near the boundary of the parameter space. For a fixed β_s value, the log-likelihood function is relatively insensitive to changes in β_h and β_v.

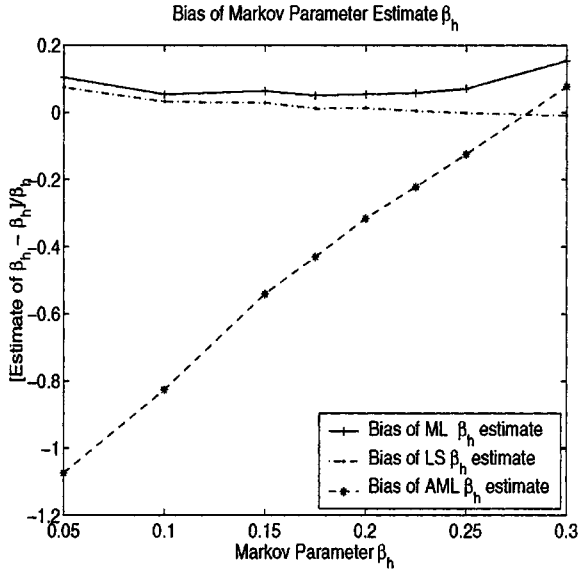


Fig. 4. Relative bias of $\widehat{\beta_h}$ computed with MC = 500: $\beta_h \in [0.05\ 0.3]$, with $\sigma^2 = 1$, $\beta_v = \beta_s = 0.1$.

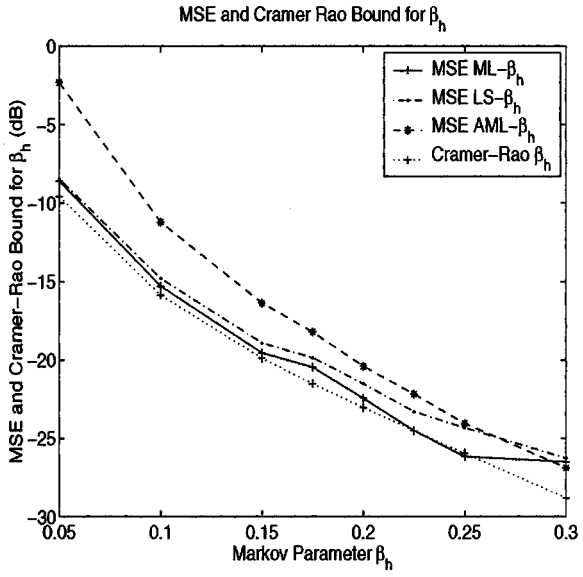


Fig. 5. Mean-square error of $\widehat{\beta_h}$ computed with MC = 500 and Cramér–Rao bound of β_h: $\beta_h \in [0.05\ 0.3]$, with $\sigma^2 = 1$, $\beta_v = \beta_s = 0.1$.

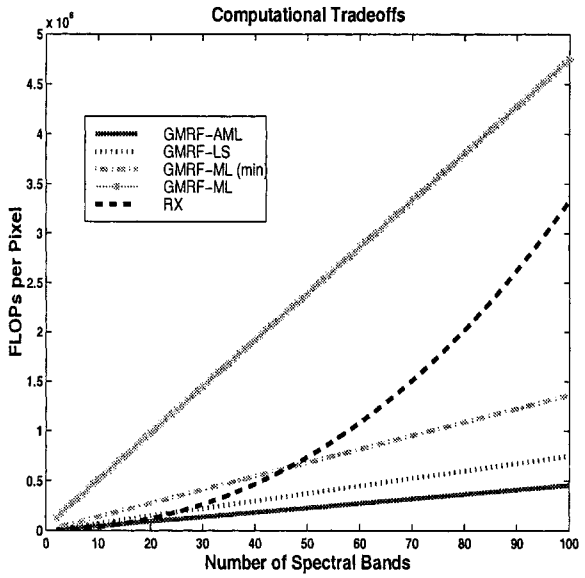
tector with the approximate maximum likelihood are close to the detection performance using the least squares estimates.

Fig. 5 shows the (sample) mean-square error (MSE) of the $\widehat{\beta_h}$ estimate as computed by the three estimation procedures, averaged again over MC = 500 Monte Carlo runs, and the Cramér–Rao bound versus the actual value of the Markov parameter β_h. These quantities are plotted in decibels computed as

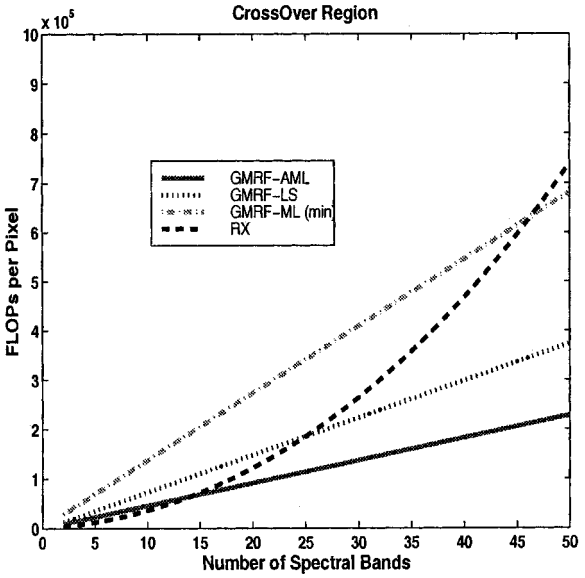
$$10\log_{10}\frac{\widehat{\sigma_i^2}}{\beta_i^2}, \qquad i = h, v, s.$$

Going from top to bottom we have: the approximate maximum-likelihood (dashed line and $*$ as marks), the least squares (dotted line and ticked by .), and the maximum-likelihood (solid line) mean-square errors, and the Cramér–Rao bound (dotted line). The ML and LS plots are very close to the Cramér–Rao bound plot, while the AML plot steadily approaches the Cramér–Rao line as we move to the right, i.e., as the field becomes more correlated the approximation underlying the AML estimates becomes better and better.

We have carried out additional studies of the bias, the mean-square error, and the Cramér–Rao bound for other choices of the values of the Markov parameters and for different values of the number of Monte Carlo runs. These studies confirm these general trends. We do not include them in this paper.



(a)



(b)

Fig. 6. Computational comparison. (a) Maximum-likelihood (ML), approximate-ML (AML), and least squares (LS), each combined with the GMRF detection framework, versus RX: the GMRF algorithms grow linearly, rather than exponentially with the number of spectral bands. (b) Zoomed-in view of the crossover region.

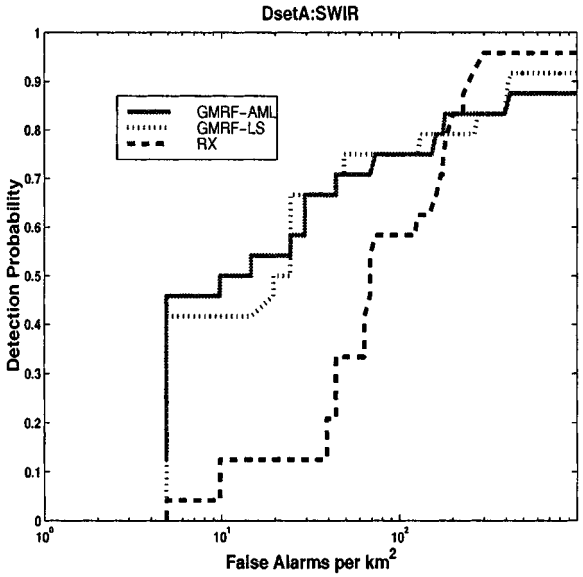
C. Computational Performance

To evaluate the computational effectiveness of the maximum-likelihood, approximate-ML, and least squares algorithms, we derive from the C-code implementation of each algorithm, an expression relating the number of floating-point operations (FLOP's) required to compute the detection statistic for one pixel in the image set to the number of spectral bands used for processing. This number of FLOP's is also dependent on the sizes of the processing, target, and Markov windows that are used. We use the detection formulation presented in Section VII. The total number of FLOP's per pixel for a specified processing and Markov window combination is then plotted as a function of the number of spectral bands used for processing.

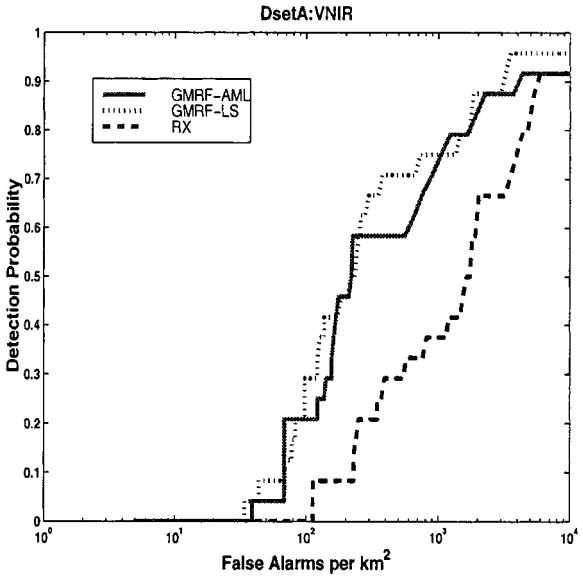
Fig. 6(a) shows the results of this analysis for the three estimation algorithms under consideration: approximate-ML (solid line), least squares (dotted line), and maximum-likelihood (dash-dot/asterisk), as well as the RX algorithm (dashed line). The maximum number of FLOP's shown on the y-axis is 5×10^6. The results are based on a 15×15 processing window, and 3×3 target and Markov windows. The RX algorithm [7] is a maximum-likelihood anomaly detection procedure that assumes spatially white clutter. The algorithm uses a binary hypothesis approach to detection, and implements a Generalized Likelihood Ratio Test (GLRT). The algorithm requires a full spectral covariance matrix to be estimated and then inverted, and was developed for multispectral sensor data. Its application to hyperspectral data has two limitations: computational difficulties arising from the matrix inversion requirement, and the suboptimal spatially white clutter assumption.

The curves in Fig. 6(a) reveal that approximate-ML is computationally superior to both maximum-likelihood and least squares. For the maximum-likelihood algorithm, the total number of FLOP's is dependent on the number of iterations it takes for the nonlinear optimization to converge. We show both the minimum possible number of FLOP's, and the total number of FLOP's when the number of convergence iterations is chosen more realistically. For all three estimation methods, the number of FLOP's increases linearly with the number of spectral bands N_b. As shown in Fig. 6(a), this is a significant improvement over other maximum-likelihood detection implementations in which the computational complexity increases with N_b^3. This exponential growth in the number of FLOP's as the number of spectral bands grows precludes the use of algorithms, such as RX, on true hyperspectral data. In contrast, the linearly increasing complexity of the approximate-ML and least squares algorithms when incorporated in our GMRF detection framework makes them viable and practical, from a computational point of view, even when using a large number of spectral bands.

Fig. 6(b) is a zoomed-in view of the region in Fig. 6(a) in which the curves of the approximate-ML, least squares, and RX algorithms cross one another. The plots indicate that the RX algorithm provides only a slight computational advantage when a small number of spectral bands is used for processing, i.e., when using multispectral imagery. However, when using more than approximately 14 spectral bands, the approximate-ML approach is computationally superior. The adequate number of spectral bands to use for processing is an issue that is currently still being analyzed.



(a)



(b)

Fig. 7. Performance on real hyperspectral data. (a) HYDICE SWIR subset. (b) HYDICE VNIR subset.

Considering the estimation results from Section VI-B and the computational results from this section together, suggests that the approximate-ML estimation algorithm is the overall best choice among the three techniques considered: approximate-ML, least squares, maximum-likelihood. Although least squares does perform slightly more reliably than approximate-ML on the simulated data, the computational improvement provided by the approximate-ML algorithm is a much more significant benefit. We focus our analysis on the approximate-ML and least squares algorithms in the context of our GMRF anomaly detector and, in the next section, present performance of these algorithms on real hyperspectral data.

VII. Detection Performance on Real Data

Our analysis of which is the most effective and efficient estimation technique to be used in the GMRF anomaly detector would not be complete without testing the methods on real hyperspectral imagery. In this section, we show performance results for the approximate-ML and least squares estimation methods in conjunction with the detection approach presented in [11], [36]. The generalized decision test is

$$t = \frac{1}{n}\sum_{q=1}^{n} (\underline{\mathrm{X}}_q - \underline{\mathrm{m}}_x)'\widehat{\Sigma_x^{-1}}(\underline{\mathrm{X}}_q - \underline{\mathrm{m}}_x) \underset{\text{target}}{\overset{\text{no target}}{\gtrless}} \lambda. \quad (51)$$

The vectors $\underline{\mathrm{X}}_q, q = 1, \ldots, n$ are the n independent field samples, $\underline{\mathrm{m}}_x$ and $\widehat{\Sigma_x^{-1}}$ are the estimates of the mean and of the inverse covariance of the clutter, and λ is a threshold. The test statistic t measures the dissimilarity between the field samples $\underline{\mathrm{X}}_q, q = 1, \ldots, n$ and the clutter; we refer the reader to [11], [36] for details and comparisons with other decision tests. In the literature the test in (51) is sometimes referred to by the misnomer "single hypothesis test," [37].

In our detection performance analysis, we use real data from the Hyperspectral Digital Imagery Collection Experiment (HYDICE) sensor[2] and the Spatially Enhanced Broadband Array Spectrograph System (SEBASS) sensor [38]. The HYDICE sensor records 210 spectral bands of data in the visible to near infrared (VNIR, 0.4 μm–1.1 μm) and the short-wave infrared (SWIR, 1.1 μm–3 μm) portions of the electromagnetic spectrum, while the SEBASS sensor records 128 bands of data in the long-wave infrared (LWIR, 5 μm–14 μm). The data from both sensors have pixels of approximately 1-m spatial resolution. In our studies below where we compare the GMRF algorithm to the RX algorithm, we restrict the number of spectral bands to a maximum of 30 for which it is still practical to apply the RX algorithm.

The first two examples for which we show results, are from the HYDICE sensor. The scene consists of 18 man-made targets, all of which sit in the open, and is 640×320 pixels in size.We run the algorithms on two subsets of bands: 22 bands from the short-wave infrared, referred to as DsetA:SWIR, and 17 bands from the visible to near infrared, referred to as DsetA:VNIR. Fig. 7 shows the receiver operating characteristic (ROC) curves [21] for the approximate-ML and least squares GMRF algorithms on the two subsets of bands. ROC curves plot the number of false alarms versus the probability of correctly detecting the targets that are present in the scene. An ideal ROC curve would be a step function which goes to 100% detection at zero false alarms. The short-wave infrared results are shown in Fig. 7(a), and the visible to near-infrared results are shown in Fig. 7(b). The third curve (dashed line) shown on the plots is for the RX algorithm [7], [9], a well-tested anomaly detection algorithm, see the discussion in Section VI-C.

As observed with the synthesized data, the approximate-ML and least squares methods perform nearly identically. In both cases, it is important to note that the two GMRF algorithms outperform the benchmark algorithm. For the number of spectral bands used in these examples, only the approximate-ML approach provides a computational advantage over RX, see Fig. 6(b).

[2]Naval Research Lab (NRL) Hydice Web Site, http://rsd-www.nrl.navy.mil/hydice/

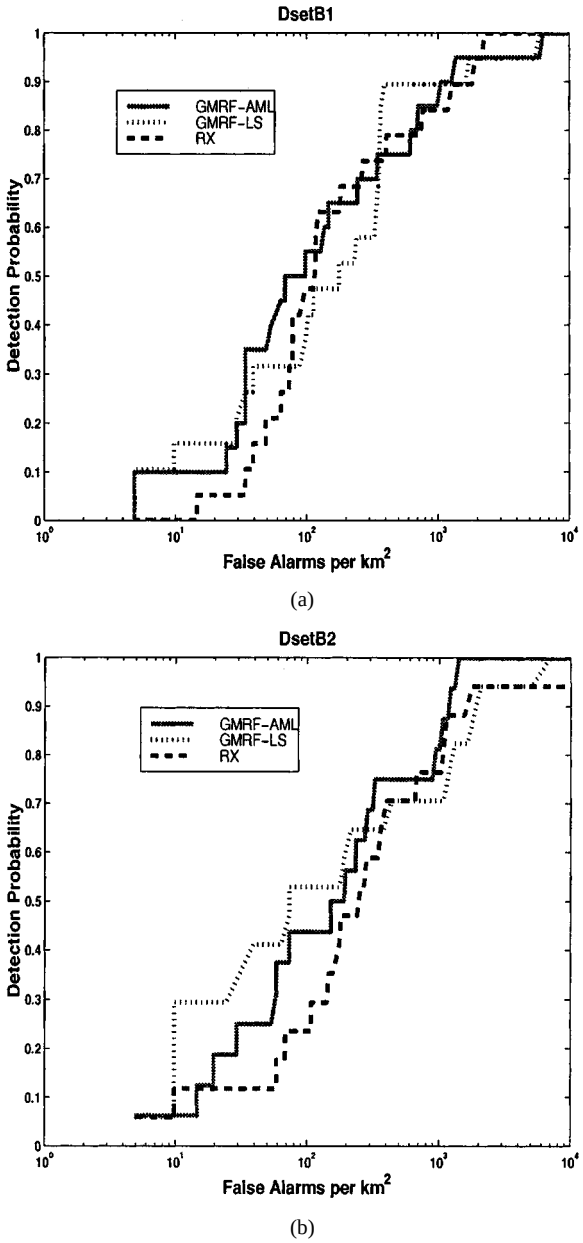


Fig. 8. HYDICE data. (a) DsetB1. (b) DsetB2.

The next two examples are again from the HYDICE sensor, but from different scenes than the last examples. In these cases, the visible to near-infrared and short-wave infrared bands are not broken into two subsets. In each example, the 210 available spectral bands are aggregated into 30 bands which span the visible to short-wave infrared. The images cover similar scenes, although the first example, referred to as DsetB1, contains 19 targets, while DsetB2 has 16 total targets. The images are again 640×320 pixels in size. The ROC results for the two hyperspectral images are shown in Fig. 8.

On DsetB1, all three algorithms show similar detection performance, while on DsetB2 the least squares method performs the best, followed by approximate-ML. Both provide slightly better performance than the RX algorithm. Using 30 bands,

TABLE II
AVERAGE PROCESSING TIME ON 30 BAND HYDICE DATA

Algorithm	Processing Time
Approximate-ML (GMRF)	35 minutes
Least Squares (GMRF)	50 minutes
RX	76 minutes

TABLE III
AVERAGE PROCESSING TIME ON 19 BAND SEBASS DATA

Algorithm	Processing Time
Approximate-ML (GMRF)	61 minutes
Least Squares (GMRF)	92.5 minutes
RX	99 minutes

both the approximate-ML and the least squares GMRF algorithms provide a significant computational improvement over RX. The average processing times of the approximate-ML, least squares, and RX algorithms on DsetB1 and DsetB2 for processing, target, and Markov windows of dimension 15×15, 3×3, and 3×3, respectively, are shown in Table II. The times shown in the table, and throughout the paper, are for C code running on a 250-MHz UltraSparc server. When using 30 spectral bands for these data sets, the RX algorithm takes twice as long as the approximate-ML GMRF approach, and 1.5 times as long as the least squares GMRF method to process one image cube of data. When the number of spectral bands is increased to 105, the differences in processing times become more drastic, e.g., the approximate-ML GMRF approach takes 4 h, while RX takes 44 h.

The last set of examples are from the SEBASS sensor. Since this is a thermal sensor, we include results on data collected at various times throughout the day. Figs. 9 and 10 show ROC results for four sets of hyperspectral imagery, referred to as DsetD1 (Night), DsetD2 (Noon), DsetD3 (Evening), and DsetD4 (Afternoon). In contrast to the other examples presented in this paper, these data sets consist of several hyperspectral image cubes. The first three sets contain five image cubes, while the fourth contains four image cubes. For clarity in the presentation of the results, we show the performance of the algorithms on a subset of the total number of available image cubes. The images contain between 30 and 40 targets, some of which are partially obscured, and the size of the images is 6100×128 pixels. We use 19 bands spanning wavelengths 8.5 to 12.5 μm. There are 78 bands available in this wavelength range, and we aggregate these bands to obtain the 19 bands used for processing.

Figs. 9 and 10 show the performance of the approximate-ML GMRF approach, the least squares GMRF method, and the RX algorithm on the SEBASS data. The "best," "worst," and "median" labels refer to the performance of the approximate-ML GMRF algorithm on the available image cubes within a particular data set. Recall that within a given SEBASS data set there are five image cubes available for analysis. We compared the ROC performance of the approximate-ML GMRF algorithm on all five image cubes by plotting the curves on one axis. Using the image cubes on which the approximate-ML GMRF algorithm showed the best, worst, and median performance, we show in Figs. 9 and 10 how the performance of the approximate-ML GMRF method compares to the performance of the least squares GMRF approach, and the performance of the RX algorithm.

As with the HYDICE examples, the approximate-ML GMRF approach and the least squares GMRF method show nearly identical performance. In each of the four SEBASS data sets, the GMRF algorithms, in general, show better detection performance than the RX algorithm at low probability of detection, but a slightly degraded performance at higher detection probabilities. Our preliminary analysis of the results indicates that the trend in the ROC curves observed in Figs. 9 and 10 is a result of there being a significant range in the size of the targets in the scenes. Fig. 11 shows ROC curves for the approximate-ML GMRF approach, the least squares GMRF method, and the RX algorithm on target types A and B for the three image cubes of DsetD2. Target type A consists of small man-made objects which, in general, extend over only 1 or 2 pixels, while target type B consists of larger man-made objects. The results indicate that the RX algorithm does better than the GMRF methods on the smaller objects (type A), but the GMRF algorithms outperform RX on the larger objects (type B). To draw general conclusions, further analysis of this trend is necessary.

A computational analysis of the SEBASS results re-emphasizes the superiority of the approximate-ML GMRF algorithm. Table III shows the average processing times over all 19 image cubes from the 4 SEBASS data sets for the approximate-ML GMRF algorithm, the least squares GMRF approach, and the RX method. Even for only 19 bands, the table shows that the RX algorithm takes 1.5 times longer to process the SEBASS HSI than the approximate-ML GMRF algorithm. The approximate-ML approach is also significantly faster than the least squares GMRF method.

VIII. SUMMARY

In this paper we have presented a clutter *adaptive* anomaly detector for hyperspectral imagery: the Gauss–Markov random field (GMRF) algorithm. Efficient processing of hyperspectral imagery requires the development of new detection methods due to the massive amount of spatial and spectral data that are captured by the sensor. The GMRF detector improves upon the benchmark multispectral anomaly detector (RX) in two critical respects: It avoids an inversion of the data covariance matrix by directly parameterizing the inverse of the covariance, and it simultaneously exploits the spatial and spectral correlation of the clutter. The critical task in the adaptive version of the GMRF anomaly detection algorithm is the estimation of the GMRF parameters defining the parameterization of the inverse of the clutter covariance. This paper focused on this issue. We have presented and analyzed three techniques for estimating the Markov parameters: the optimal maximum-likelihood approach, a least squares method, and an approximate maximum-likelihood procedure.

The three estimation techniques were evaluated along three main criteria: estimation performance; computational cost; and detection performance of the adaptive GMRF anomaly detector as evaluated with real data. The estimation accuracy

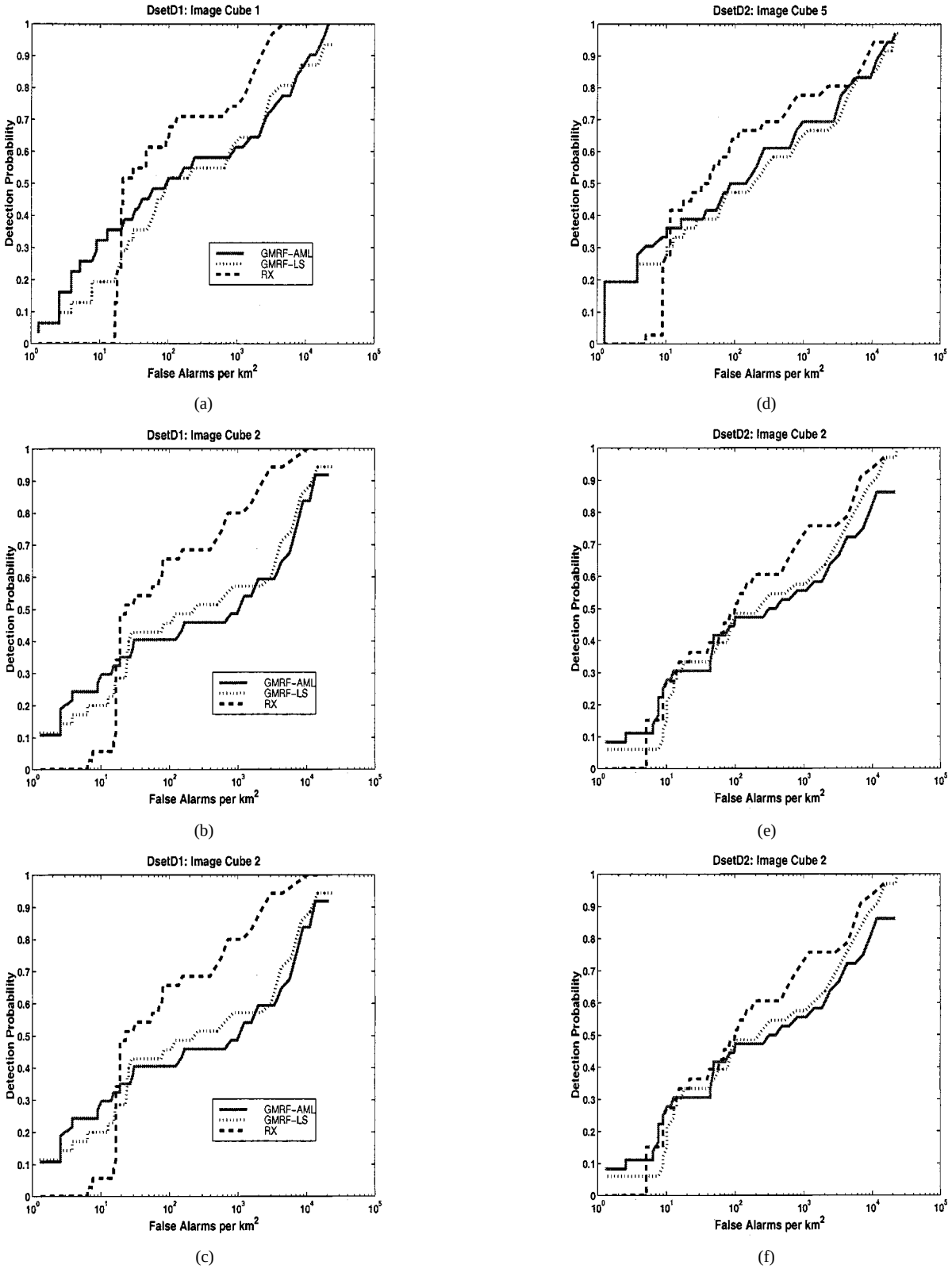


Fig. 9. SEBASS data. (a) DsetD1: "best." (b) DsetD1: "worst." (c) DsetD1: "median." (d) DsetD2: "best." (e) DsetD2: "worst." (f) DsetsD2: "median."

was studied by computing analytically and numerically the Cramér–Rao bounds for the Markov parameters, and by Monte Carlo simulation studies. The Monte Carlo experiments show that the three estimates are well-behaved: they have relatively small biases, and their sample mean-square errors track closely the Cramér–Rao bound. An interesting observation is that, with highly correlated fields, the Monte Carlo based study shows that both in terms of the bias and the mean-square error the approximate maximum-likelihood estimation error performance improves as the Markov parameters become closer to the boundary of the parameter space, i.e., as the random field becomes highly correlated.

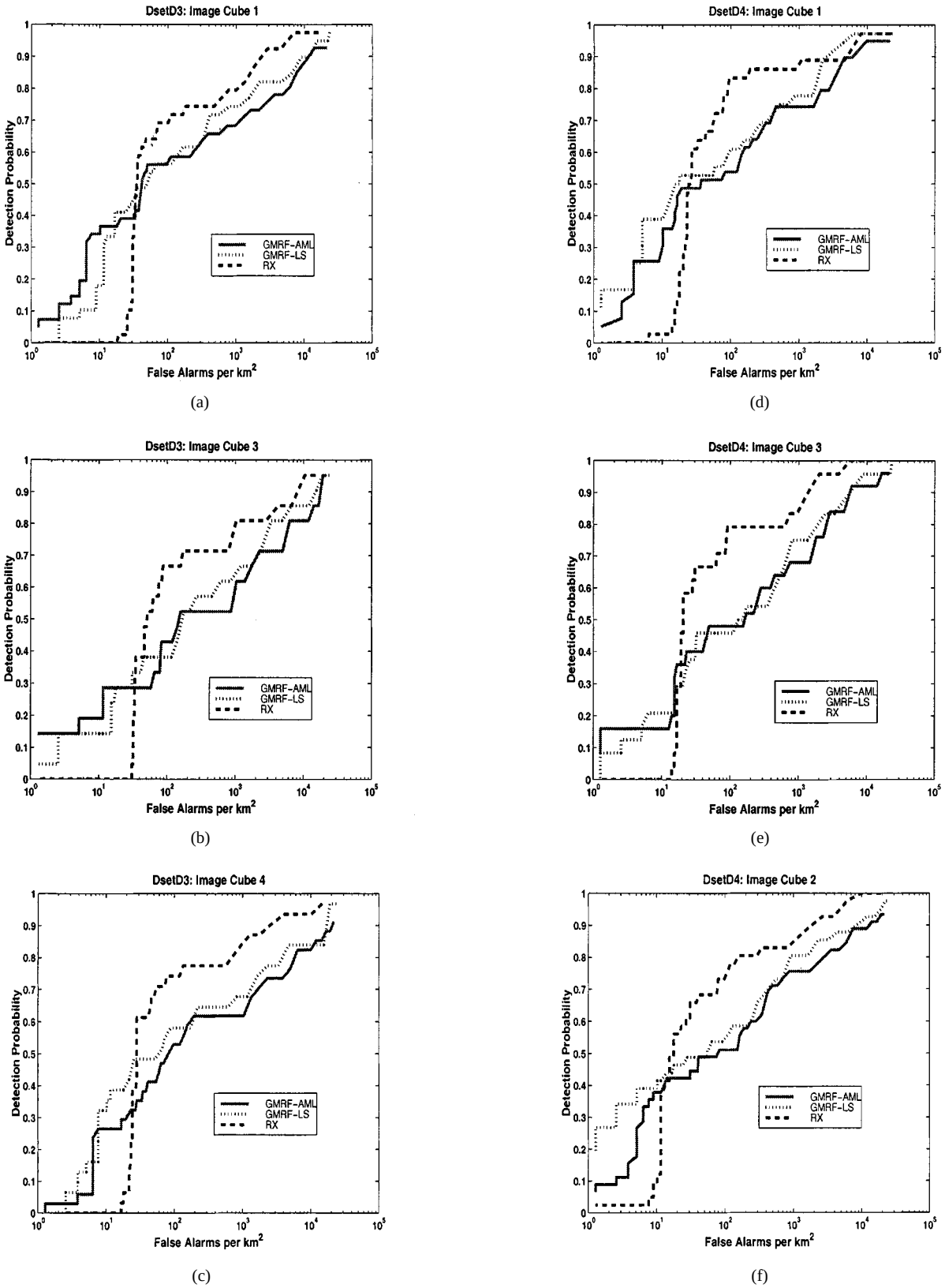


Fig. 10. SEBASS data. (a) DsetD3: "best." (b) DsetD3: "worst." (c) DsetD3:"median." (d) DsetD4: "best." (e) DsetD4: "worst." (f) DsetD4: "median."

We computed the computational cost of the three estimation techniques and concluded that the approximate maximum likelihood has a significant computational advantage over the other two.

The paper reported on the extensive testing results of our adaptive GMRF anomaly detector with real hyperspectral imagery. We focus on the combination of the approximate maximum-likelihood estimation algorithm and of the least

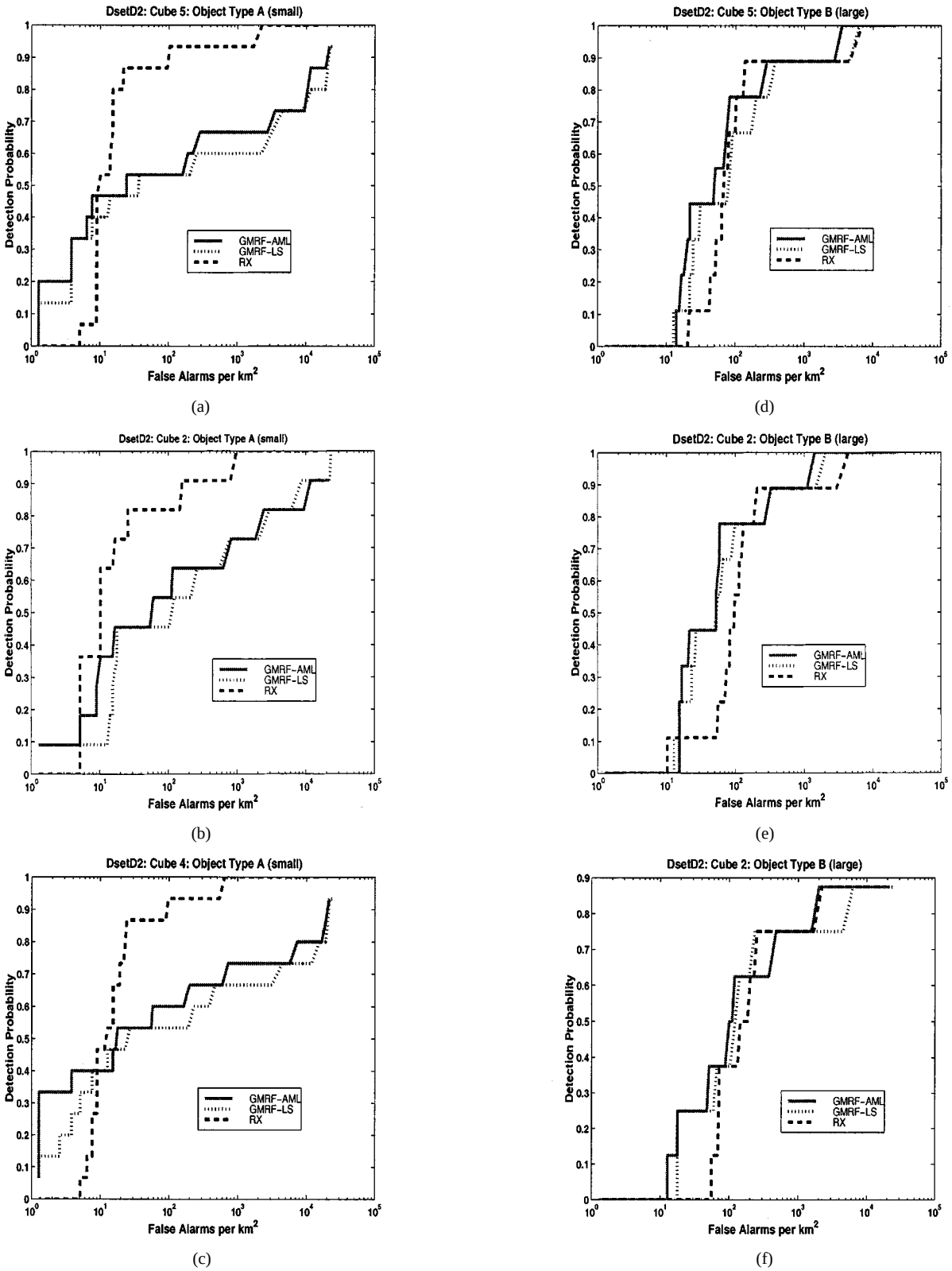


Fig. 11. Performance by object type. object type A (small) versus object type B (large). (a) DsetD2: "best": type A. (b) DsetD2: "worst": type A. (c) DsetD2: "median": type A. (d) DsetD2: "best": type B. (e) DsetD2: "worst": type B. (f) DsetD2: "median": type B.

squares estimation procedure with the generalized detection scheme presented in [11]. We evaluated the effectiveness of the approximate-ML adaptive GMRF anomaly detector and the least squares adaptive GMRF algorithm in terms of detection performance on real hyperspectral imagery, as well as in terms of computational complexity. The HYDICE and SEBASS performance results presented in the paper highlight the approximate-ML GMRF algorithm as a promising detector for hyperspectral data, particularly with respect to its computational performance. The approximate-ML GMRF algorithm,

in general, provides comparable performance to the RX algorithm, and, in certain instances, may provide better detection performance, e.g., on "large" targets in the SEBASS data, while providing a significant computational advantage, even when the full set of spectral bands has been reduced through aggregation. For instance, the 128 bands of the SEBASS data were reduced to 19 bands, and the RX algorithm still took 1.5 times longer than the approximate-ML GMRF detector to process one image cube. The computational advantage of GMRF over RX is rooted in the fact that the complexity of the GMRF algorithm increases linearly with the number of spectral bands N_b rather than with N_b^3. It is this property of the approximate-ML GMRF algorithm that makes it a viable adaptive detector for full-spectral-band hyperspectral sensor imagery; a task that is beyond the feasibility of the RX algorithm. The encouraging overall performance of the approximate-ML adaptive GMRF detector on these preliminary data sets has led to it being considered as a possible alternative detection algorithm for a new hyperspectral imagery sensor system being developed under the Adaptive Spectral Reconnaissance (ASR) program funded by the Sensor Technology Office of DARPA.

ACKNOWLEDGMENT

The authors wish to thank Dr. Ray Johnson, Dr. Wayne Summers, and Dr. Bob Best from Science Applications International Corporation in Arlington, VA, as well as the Multi-Sensor Exploitation Testbed (MSET) and ASR programs in DARPA/STO, run by LtCol. Dr. Brad Tousley, for their support of this work, and especially for providing the real hyperspectral data sets used for analysis. Finally, the authors acknowledge the careful reading and suggestions of three anonymous reviewers and of Prof. Joseph A. O'Sullivan.

REFERENCES

[1] T. M. Lillesand and R. W. Kiefer, *Remote Sensing and Image Interpretation*. New York: Wiley, 1994.

[2] J. B. Campbell, *Introduction to Remote Sensing*, 2nd ed. New York: Guilford, 1996.

[3] R. A. Schowengerdt, *Remote Sensing, Models and Methods for Image Processing*, 2nd ed. San Diego, CA: Academic, 1997.

[4] A. Stocker, X. Yu, E. Winter, and L. Hoff, "Adaptive detection of subpixel targets using multi-band frame sequences," *SPIE: Signal and Data Processing of Small Targets*, vol. 1481, pp. 156–169, Apr. 1991.

[5] A. D. Stocker, A. Oshagan, J. H. Seldin, J. N. Cederquist, and C. R. Schwartz, "Analysis of infrared multi-spectral target/background field measurements," *SPIE: Signal and Data Processing of Small Targets*, vol. 2235, pp. 148–161, 1994.

[6] M. T. Eismann, J. H. Seldin, C. R. Schwartz, J. R. Maxwell, K. K. Ellis, J. N. Cederquist, A. D. Stocker, A. Oshagan, R. O. Johnson, W. A. Shaffer, M. R. Surette, M. J. McHugh, A. P. Schaum, and L. B. Stotts, "Target detection in desert backgrounds: Infrared hyperspectral measurements and analysis," *SPIE: Signal and Data Processing of Small Targets*, vol. 2561, pp. 80–97, 1995.

[7] X. Yu, L. E. Hoff, I. S. Reed, A. M. Chen, and L. B. Stotts, "Automatic targe detection and recognition in multiband imagery: A unified ML detection and estimation approach," *IEEE Trans. Image Processing*, vol. 6, pp. 143–156, Jan. 1997.

[8] A. Margalit, I. S. Reed, and R. M. Gagliardi, "Adaptive optical target detection using correlated images," *IEEE Trans. Aerosp. Electron. Syst.*, vol. AES-21, pp. 394–405, June 1985.

[9] I. S. Reed and X. Yu, "Adaptive multiple-band CFAR detection of an optical pattern with unknown spectral distribution," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 38, no. 10, pp. 1760–1770, Oct. 1990.

[10] J. Y. Chen and I. S. Reed, "A detection algorithm for optical targets in cluster," *IEEE Trans. Aerosp. Electron. Syst.*, vol. AES-23, pp. 46–59, Feb. 1987.

[11] S. M. Schweizer and J. M. F. Moura, Efficient detection in hyperspectral imagery, Mar. 1999, submitted for publication.

[12] J. Besag, "Spatial interaction and the statistical analysis of lattice systems," *J. Roy. Statist. Soc.*, vol. B36, no. 2, pp. 192–236, 1974.

[13] P. A. P. Moran, "A Gaussian Markovian process on a square lattice," *J. Appl. Probab.*, vol. 10, pp. 54–62, 1973.

[14] S. Geman and D. Geman, "Stochastic relaxation, Gibbs distribution, and Bayesian restoration of images," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. PAMI-6, pp. 721–741, Nov. 1984.

[15] B. R. Hunt and T. M. Cannon, "Nonstationary assumptions for gaussian models of images," *IEEE Trans. Syst., Man, Cybern.*, vol. 6, pp. 876–882, Dec. 1976.

[16] J. W. Woods, "Two-dimensional discrete Markovian fields," *IEEE Trans. Inform. Theory*, vol. IT-18, pp. 232–240, Mar. 1972.

[17] J. M. F. Moura and N. Balram, "Recursive structure of noncausal Gauss-Markov random fields," *IEEE Trans. Inform. Theory*, vol. 38, pp. 334–354, Mar. 1992.

[18] ——, "Statistical algorithms for numerical Gauss-Markov fields," in *Handbook of Statistics*, N. Bose and N. Rao, Eds. Amsterdam, The Netherlands: North-Holland, July 1993, vol. X, ch. 15, pp. 623–691.

[19] A. K. Jain, *Fundamentals of Digital Image Processing*. Englewood Cliffs, NJ: Prentice Hall, 1989, ch. 2: Two Dimensional Systems and Mathematical Preliminaries.

[20] J. R. Magnus and H. Neudecker, *Matrix Differential Calculus*. New York: Wiley, 1988, ch. 2: Kronecker Products, the vec Operator and the Moore-Penrose Inverse.

[21] H. L. Van Trees, *Part I: Detection, Estimation, and Modulation Theory*. New York: Wiley, 1968, ch. 2: Classical Detection and Estimation Theory.

[22] G. Strang, *Linear Algebra and Its Applications*, 3rd ed. Orlando, FL: Harcourt, Brace, Jovanovich, 1988.

[23] N. Balram and J. M. F. Moura, "Noncausal Gauss-Markov random fields and parameter structure and estimation," *IEEE Trans. Inform. Theory*, vol. 39, pp. 1333–1355, July 1993.

[24] A. K. Jain, "A sinusoidal family of unitary transforms," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. PAMI-1, pp. 356–365, Oct. 1979.

[25] C. F. Fischer and R. A. Usmani, "Properties of some tridiagonal matrices and their application to boundary value problems," *SIAM J. Numer. Anal.*, vol. 6, pp. 127–142, Mar. 1969.

[26] M. Bruno and J. M. F. Moura, "DCT/DST and Gauss-Markov fields: Conditions for equivalence," *IEEE Trans. Signal Processing*, vol. 46, pp. 2571–2574, Sept. 1998.

[27] T. W. Andresen, "Statistical inference for covariance matrices which are linear combinations or whose inverses are linear combinations of given matrices," in *Essays in Probability and Statistics*, C. R. Rao *et al.*, Eds. Raleigh: Univ. North Carolina Press, 1970, ch. 1, pp. 1–24.

[28] J. P. Burg, D. G. Luenberger, and D. L. Wenger, "Estimation of structured covariance matrices," *Proc. IEEE*, vol. 70, pp. 963–974, Sept. 1982.

[29] S. Bose and A. O. Steinhardt, "A maximum invariant framework for adaptive detection with structured and unstructured covariance matrices," *IEEE Trans. Signal Processing*, vol. 43, pp. 2164–2175, Sept. 1995.

[30] H. Li, P. Stoica, and J. Li, "Computationally efficient maximum likelihood estimation of structured covariance matrices," *IEEE Trans. Signal Processing*, vol. 47, pp. 1314–1323, May 1999.

[31] L. V. S. Boyd and S.-O. Wu, "Determinant maximization with linear matrix inequality constraints," *SIAM J. Matrix Anal. Applic.*, vol. 19, pp. 499–533, 1998.

[32] W. H. Press, W. T. Vetterling, S. A. Teukolsky, and B. P. Flannery, *Numerical Recipes in C: The Art of Scientific Computing*, 2nd ed. Cambridge, U.K.: Cambridge Univ. Press, 1992, ch. 10.

[33] S.-T. Bow, *Pattern Recognition and Image Preprocessing*. New York: Marcel Dekker, 1992.

[34] L. L. Scharf, *Statistical Signal Processing: Detection, Estimation, and Time Series Analysis*. Reading, MA: Addison-Wesley, 1991.

[35] C. F. Ferrara, "Adaptive spatial/spectral detection of subpixel targets with unknown spectral characteristics," *SPIE: Signal and Data Processing of Small Targets*, vol. 2235, pp. 82–93, 1994.

[36] S. M. Schweizer and J. M. F. Moura, "ML-detection in hyperspectral imagery: A GMRF modeling approach," *SPIE: Algorithms for Multispectral and Hyperspectral Imagery V*, Apr. 1999.

[37] K. Fukunaga, *Statistical Pattern Recognition*, 2nd ed. San Diego, CA: Academic, 1990.

[38] *Hydra Data Collection Data Distribution Manual* , Contact Mike Eismann, Air Force Research Lab. (AFRL), Nov. 1998.

[39] J. P. Burg, D. G. Luenberger, and D. L. Wenger, *Nonlinear Stochastic Problem*, R. S. Bucy and J. M. F. Moura, Eds. Amsterdam, The Netherlands: Reidel, May 1982.