Detection of Subpixel Anomalies in Multispectral Infrared Imagery Using an Adaptive Bayesian Classifier

Edward A. Ashton

***Abstract*— The detection of subpixel targets with unknown spectral signatures and cluttered backgrounds in multispectral imagery is a topic of great interest for remote surveillance applications. Because no knowledge of the target is assumed, the only way to accomplish such a detection is through a search for anomalous pixels. Two approaches to this problem are examined in this paper. The first is to separate the image into a number of statistical clusters by using an extension of the well-known k-means algorithm. Each bin of resultant residual vectors is then decorrelated, and the results are thresholded to provide detection. The second approach requires the formation of a probabilistic background model by using an adaptive Bayesian classification algorithm. This allows the calculation of a probability for each pixel, with respect to the model. These probabilities are then thresholded to provide detection. Both algorithms are shown to provide significant improvement over current filtering techniques for anomaly detection in experiments using multispectral IR imagery with both simulated and actual subpixel targets.**

***Index Terms*—Classification, multispectral, segmentation, subpixel, target detection.**

I. Introduction

SEVERAL recent studies have demonstrated the usefulness of taking information from multiple spectral bands when searching for targets that are poorly resolved spatially [1]–[3]. The majority of work in this area, however, has focused on the detection of targets with known spectral properties. While this knowledge simplifies the problem, it also renders it less realistic. It is assumed in these studies that laboratory or field measurements of target spectra may be used as filter vectors for target detection. However, there are several factors that make this difficult to put into practice. First, atmospheric interference may vary under different weather conditions, changing the measured spectral signature of the target. Second, in subpixel targets, the target spectrum is mixed with the spectra of random background elements. Third, and most importantly, in a realistic situation, knowledge of precisely what type of paint or camouflage the target is using may not be available.

These conditions lead us to seek a solution in which the statistical characterizations are based entirely on the data under consideration rather than on some previously measured spectra. Because targets are assumed to be very sparsely represented in a scene and their locations are unknown, they cannot be characterized in this way. However, it is possible to derive a set of background characteristics and to then search for pixels that appear to be anomalous, with respect to the background. Clearly, a definitive identification of a target cannot be made through a search for anomalies. However, such a search, which requires very low spatial resolution and can therefore be conducted over a wide search area, can be extremely useful as a cuing device for another sensor with much higher spatial resolution. It is in this role that the algorithms described here may be useful.

Identification of poorly resolved targets in single-band imagery has been demonstrated by Soni through the use of spatial decorrelation (whitening) followed by energy detection [4]. Further work by Reed and Yu with multispectral imagery has shown that a combination of spectral decorrelation and a rotation-invariant-spatial matched filter (MF) [5] provides improved detection over either algorithm alone when knowledge of target shape is available and that a spatial-spectral MF provides even greater improvement when the target spectrum is known as well [6]. However, in the case of subpixel targets, the spatial MF kernel reduces to an impulse function and these algorithms reduce to simple spectral filtering.

Spectral decorrelation filtering has been shown to be an effective tool for the identification of anomalies in multispectral imagery. However, this method is based upon the assumption that all background pixels within some arbitrarily chosen window about the pixel under consideration should form a single multivariate Gaussian distribution. While this is true in certain situations, it is more generally the case that several different terrain types will occupy at least some of the pixels within this window unless the window is made unreasonably small. Different terrain types usually have different spectral signatures and so form at least partially disjoint distributions in spectral space. Clustering or classification algorithms have frequently been used to determine terrain types in single-band surveillance imagery [7]. It is possible to extend such clustering techniques into the multispectral domain with certain modifications in the interest of saving computation time. These modifications are detailed in a following section. If the terrain type of each pixel is known, it should be possible to decorrelate each of these distributions separately, presumably with improved results. Alternatively, it should be possible to find the spectral means and variances of each distribution. Once this is done, the probability of each pixel can be calculated, with respect to some *a priori* model for pixel and region distribution. Each of these approaches is explored in this paper.

Manuscript received September 3, 1996; revised April 30, 1997.

The author is with Code 5621, Naval Research Laboratory, Washington, DC 20375 USA (e-mail: ashton@dynasun.nrl.navy.mil).

Publisher Item Identifier S 0196-2892(98)00735-9.

The following section describes in detail the models used for pixel and region distribution. Section III describes the adaptive clustering algorithm, and Section IV details the validation experiments that were carried out on multispectral IR data containing both implanted and actual military targets. A discussion of the results of these experiments is presented in Section V.

II. Background Modeling

A. k-Means Clustering

The first spectral clustering algorithm used here is a simple extension into multiple dimensions of the well-known k-means algorithm, nearly identical in form to the so-called LBG algorithm [8]. We begin with the assumption that each pixel in a multispectral scene will tend to fall into one of several clusters or bins in spectral space. For example, trees, grasses, and bare soil may form three partially overlapping but largely separate bins. The data within each bin is assumed to take on a multivariate Gaussian distribution. The clustering algorithm is able to identify the centroid of each bin, operating on a training sequence that may consist of the entire image under consideration, or may be limited to some subset of the available pixels. Once this is done, a residual vector, given by

$$r_{i,k} = y_{i,k} - c_k \tag{1}$$

is calculated at each pixel, where $y_{i,k}$ is the pixel vector at site i, which is assigned to bin k, and c_k is the centroid vector for bin k. Each bin is then independently decorrelated by using an adaptive whitening filter of the sort described in [5], and the results are thresholded to achieve detection. This relatively simple algorithm was tested on a number of IR data sets with both imaged and implanted targets and has been shown to provide significant improvement over spectral filtering alone in all cases. Detailed results of these experiments are given in Section IV.

B. Bayesian Clustering

The second clustering algorithm that is examined is similar in concept to the one outlined above. However, in this case, a Bayesian probabilistic model for the background is formed. Given the observed image data y, the most likely classification x at each pixel is found according to the formula

$$P(x|y) \propto P(y|x)P(x) \tag{2}$$

where $P(y|x)$ is given by a statistical model for pixel distribution and $P(x)$ is given by a probabilistic model for region distribution. In this implementation, the statistical model is a multivariate Gaussian distribution and the probabilistic model is a Gibbs random field (GRF) [9], [10]. Initial parameters for the statistical model are estimated from an LBG clustering of some training sequence of data, as outlined above. Local adaptivity is then added through a reestimation of local means and variances by using a shrinking estimation window, in the manner demonstrated for single-band data in [11]–[13]. Once the model is formed, it is a simple matter to calculate the probability of each pixel, with respect to its classification. Detection is achieved by thresholding the resultant scalars, with the least-probable pixels being the most likely targets.

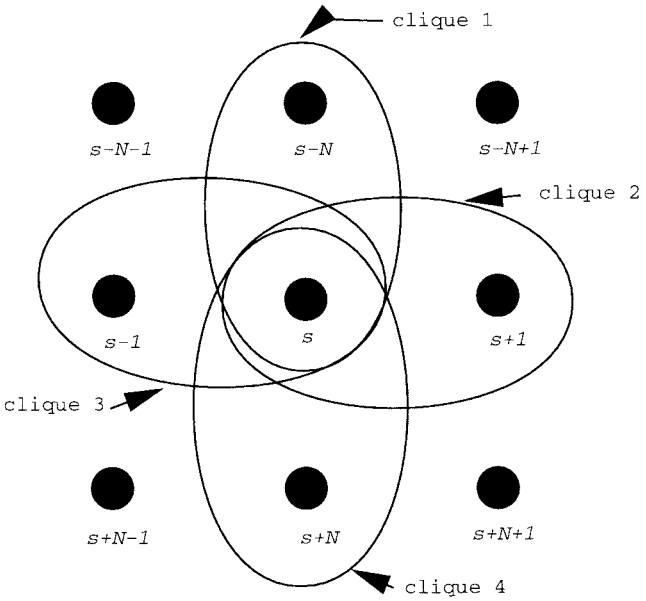


Fig. 1. Diagram shows the neighborhood of a pixel located at site s. The pixel is a member of four two-pixel cliques, one with each of its nearest neighbors. It is also a member of a one-pixel clique with itself, but as this adds no information it is disregarded.

Stochastic relaxation classifiers of this type have rarely been applied to multispectral data, primarily because they tend to be extremely computationally intensive. However, in this implementation, computation time is reduced greatly by making use of a *pixelwise stopping rule*, as described by Watanabe *et al.* in [14] and by achieving stochastic relaxation through Besag's iterated conditional modes (ICM) [15]. Dubes and Jain [16] have reported that the typical ratios of compute time for ICM, maximum posterior marginals (MPM), and simulated annealing (SA) are $1:37:6000$. With these modifications, processing time on a Silicon Graphics Indigo workstation for a 100×100-pixel six-band image is less than 4 min. Moreover, several further modifications allow a derivative of this algorithm to be applied to a real-time data stream with a minimal reduction in performance.

In order to optimize the classification of the image under consideration according to the maximum *a posteriori* probability (MAP) criterion [9], it is necessary to obtain functional forms for both $P(x)$ and $P(y|x)$. As $P(x)$ has no dependence upon the observed image data y, it must derive entirely from some *a priori* model for region distribution. This term is intended to introduce some spatial context into the classification process. Bouman [17] and others have accomplished this sort of modeling by using the Markov random field (MRF), which is derived from the one dimensional Markov chain. However, this model is defined in terms of local properties, making the derivation of a global joint distribution difficult. We therefore make use of the similar, but more mathematically tractable GRF.

Assume that a random field X is defined over an $N \times N$ lattice of sites denoted by S, with the individual sites (pixels)

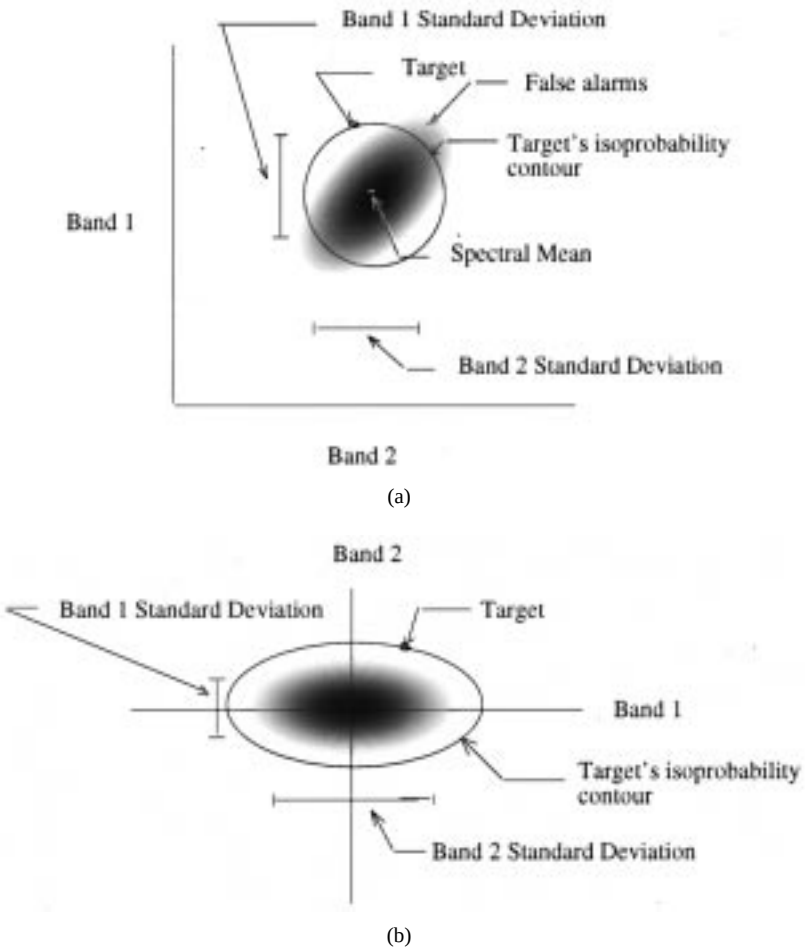


Fig. 2. (a) Typical single-class two-band data distribution. Distribution does take the form of a multivariate Gaussian, but it cannot be accurately characterized by the means and variances calculated in bands 1 and 2 because the major axes of the distribution are not aligned with the band axes. (b) Same distribution, transformed through the Hotelling transform. The variances calculated in bands 1 and 2 now accurately characterize the distribution, allowing detection of the target.

denoted by $s_1 \cdots s_{N^2}$. For the purposes of classification, each site on X may take on any integer value from one to k, where k is the number of classes (terrain types) into which the image is classified. Let ω be a particular realization of X. In order to define the random field, it is necessary first to define a neighborhood system on the lattice S. The neighborhood system determines which pixels on the lattice will directly affect the classification of site s. In this work, a first-order or four-point neighborhood system is used, so that only pixels that share a side with site s are considered its neighbors.

A *clique* is defined as a set of sites on S, such that all points are mutual neighbors. In a four-point system, there are four two-point cliques, each of which contains s and one of its four neighbors. This concept is illustrated in Fig. 1.

The GRF provides a global model for the distribution of regions in an image, specifying a probability-density function (pdf) of the form

$$P(X = x) = e^{-U(x)}/Z \tag{3}$$

where $U(x)$ is the Gibbs potential, given by the summation over all cliques C

$$U(x) = \sum_{C} V_C(x) \tag{4}$$

and $V_C(x)$ is the clique potential, given by $-\beta$, if all points in the clique are classified identically, and $+\beta$, otherwise. β is a weighting factor that determines the relative importance of the two terms in the Bayesian formulation, and Z is a normalizing constant called the partition function, given by

$$Z = \sum_{\omega} e^{-U(\omega)}. \tag{5}$$

Because Z is the sum of the numerator exponents over all possible realizations of X, it is generally impossible to compute. This intractability prevents the realization of a GRF using direct calculation, thus requiring the use of stochastic relaxation methods, such as ICM. Using the Gibbs formulation, it is possible to calculate the global probability of a given image classification based on the clique potentials, which depend solely on the local properties of the pixels. A more detailed examination of random-field modeling may be found in [15] and [16].

Returning to (2), it is necessary now to find a functional form for $P(y|x)$. This is derived from the assumption of a multivariate Gaussian distribution of each pixel class in spectral space. Under the additional assumption that each class has a slowly varying mean in each band, given at pixel s by

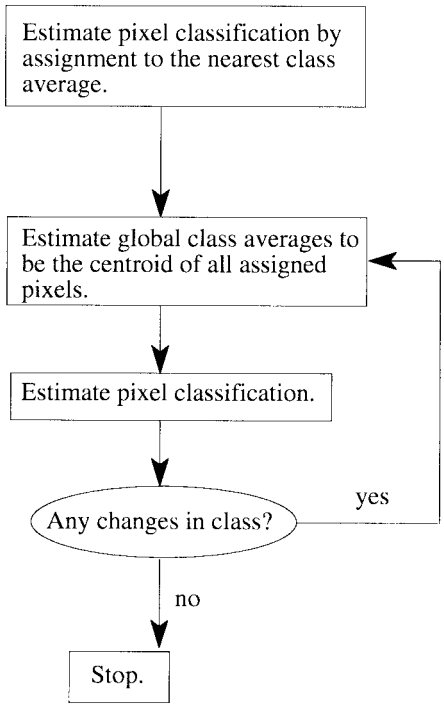


Fig. 3. k-means (LBG) clustering algorithm. Although this algorithm requires some initial estimate of class averages, this estimate need not be overly accurate. Initial accuracy will only affect speed of convergence.

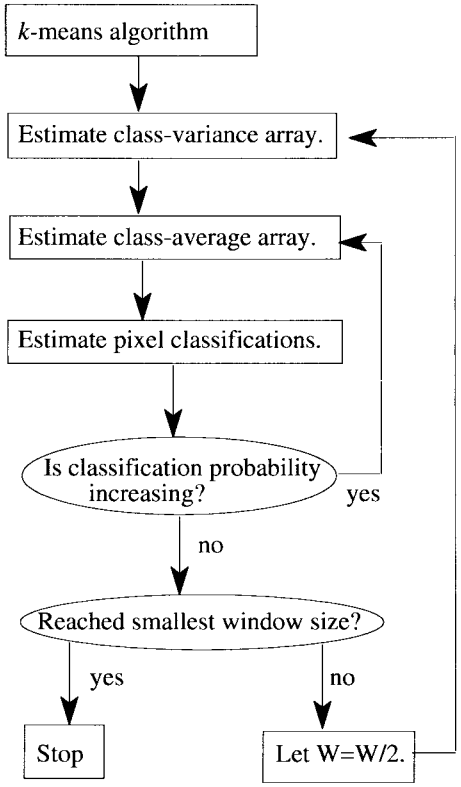


Fig. 4. Flowchart for modified adaptive clustering. Interior loop represents standard ICM. Note that estimation of the class-variance array is outside this loop.

$\mu_{(x,s,\lambda)}$, where λ refers to spectral wavelength, and a slowly varying standard deviation in each band, given at pixel s by $\sigma_{(x,s,\lambda)}$, the global probability of the observed image cube y, given the current classification lattice x, is given by

$$P(y|x) \propto \prod_S \prod_\lambda \left\{ \frac{1}{\sigma_{(x,s,\lambda)}} \right\} \exp - \sum_S \sum_\lambda \frac{1}{\sigma^2_{(x,s,\lambda)}} \cdot [y_{(s,\lambda)} - \mu_{(x,s,\lambda)}]^2. \tag{6}$$

Substituting into (2) and summing exponents obtains

$$P(x|y) \propto \prod_S \prod_\lambda \left\{ \frac{1}{\sigma_{(x,s,\lambda)}} \right\} \cdot \exp - \left\{ \sum_S \sum_\gamma \frac{1}{\sigma^2_{(x,s,\lambda)}} \cdot [y_{(s,\lambda)} - \mu_{(x,s,\lambda)}]^2 + \sum_C V_C(x) \right\}. \tag{7}$$

We wish to maximize this function or, equivalently, to maximize its natural logarithm, given by

$$\ln[P(x|y)] \propto - \sum_S \sum_\lambda \left\{ \ln[\sigma_{(x,s,\lambda)}] + \frac{1}{\sigma^2_{(x,s,\lambda)}} \cdot [y_{(s,\lambda)} - \mu_{(x,s,\lambda)}]^2 \right\} - \sum_C V_C(x). \tag{8}$$

This is accomplished in an iterative fashion by using ICM.

There is one additional minor complication that relates to the Bayesian model. The major axes of the data distributions in spectral space tend not to be aligned with the band axes. This situation is illustrated in Fig. 2(a) for a typical single-class, two-band data distribution. This data does fall into a multivariate Gaussian distribution, with a large variance in the $+45°$ direction and a smaller variance in the $-45°$ direction. However, the variances, as measured in bands 1 and 2, are identical. The probability of each pixel will therefore be proportional to its Euclidean distance from the spectral mean, and as a result, the target, which is separable from the background, will appear to be less improbable than many background pixels at the extrema of the distribution's long axis.

In order to take full advantage of the power of the Bayesian model, it is necessary to transform the data in such a way that it takes on the distribution shown in Fig. 2(b), in which the major and minor axes of the distribution are aligned with the band axes. This transformation, which may be easily and quickly accomplished through use of the Hotelling transform [18], essentially provides separability; i.e., it allows the distribution to be accurately characterized by an independent mean and variance in each band. The Hotelling transform maps the observed data vectors y_i into transformed vectors w_i through the formula

$$w_i = A(y_i - m_y) \tag{9}$$

where A is a matrix whose rows are formed from the eigenvectors of the covariance matrix of y, ordered such that

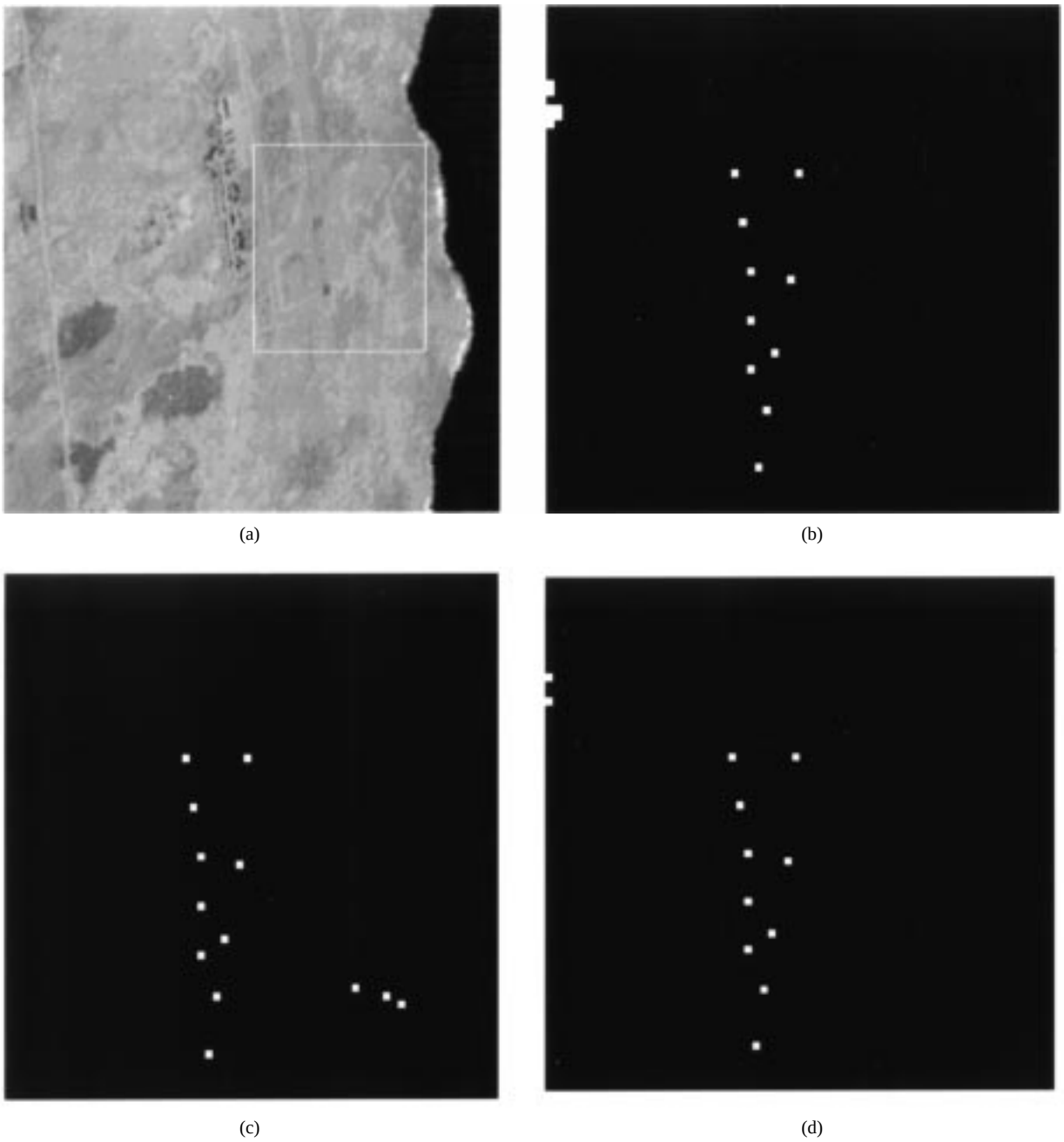
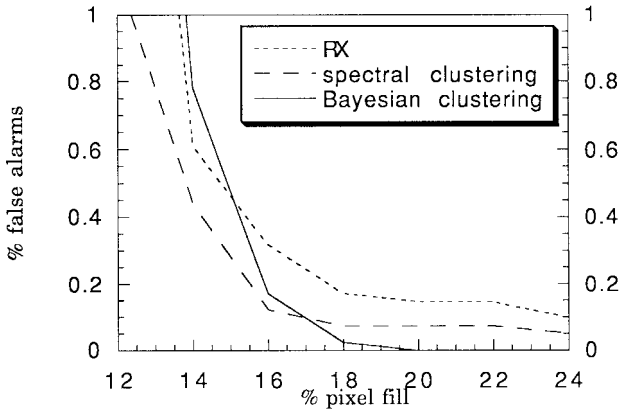


Fig. 5. (a) 200 × 200-pixel section from band 1 of the TIMS Kona image cube. Contains ten targets implanted with a pixel-fill factor of 18%. (b) Smaller section highlighted in (a), processed using spectral decorrelation filtering and thresholded at 100% detection. FAR is 0.17%. (c) Processed using spectral clustering followed by bin decorrelation. FAR is 0.07%. (d) Processed using Bayesian clustering. FAR is 0.04%.

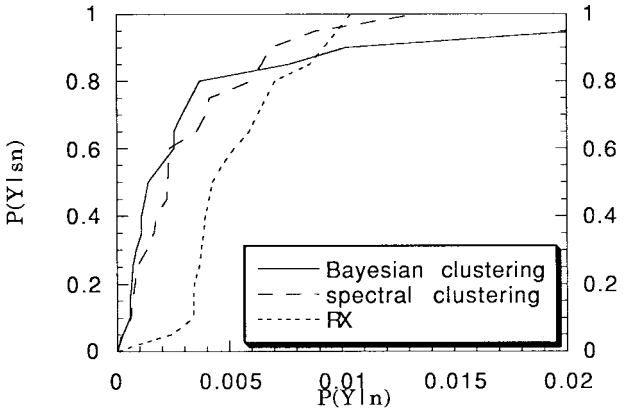
the first row is the eigenvector corresponding to the largest eigenvalue and the last row is the eigenvector corresponding to the smallest eigenvalue and m_y is the spectral mean vector of y. This operation is time consuming only if there are a very large number of bands under consideration. For the number of bands typically seen in IR cameras (6–12), the Hotelling transform adds little to the computation time of the algorithm.

III. Algorithms for Clustering and Detection

As has been previously mentioned, the algorithm used to achieve spectral clustering is a simple extension into multiple dimensions of the well-known k-means algorithm. This algorithm provides alternating estimation of the class averages (or bin centroids) and pixel classification. A flowchart for this algorithm is given in Fig. 3. Note that this flowchart assumes some initial estimate of bin centroids. This estimate does not need to be accurate for the algorithm to converge to an optimal solution. However, an initial estimate that is close to the final solution will require fewer iterations to converge than one that is far from the correct answer. Such an estimate can be found through a modification of the splitting algorithm [8]. The original splitting algorithm reaches an initial bin-centroid estimate by beginning with a single bin that contains the entire set of training data. Additional bins are produced by perturbing all centroids in the current set by some



(a)



(b)

Fig. 6. (a) Plot of pixel fill versus FAR at 100% target detection for the TIMS Kona data set. (b) ROC curves at 14% pixel-fill factor. $P(Y|sn)$ denotes probability of detection, while $P(Y|n)$ denotes probability of false alarm. Note that both clustering algorithms provide significant improvement over RX filtering alone.

small value ε, doubling the current number of bins. Optimal centroids are then calculated at the new number of bins. This process is repeated until the desired number of bins is reached. While splitting by the LBG process is optimal in terms of performance, it is also very time consuming. We save time by allowing only a limited number of iterations at each level until the desired number of bins is reached. The algorithm is then allowed to run to its conclusion on the final set of centroids. This modification provides results that are generally identical to those achieved by using the full splitting algorithm, with considerable savings in computation time.

Once clustering is accomplished, a spectral correlation matrix R_k is calculated for each of the k bins. Local adaptivity may be added by calculating R_k using a local estimation window at a series of grid points evenly spaced through the image. Values of R_k at pixels between grid points are then obtained through bilinear interpolation of the grid points. This technique is used to provide local adaptivity for the Bayesian classification algorithm, which will be described shortly. Once the set of correlation matrices has been calculated, a scalar matrix z is formed through the transformation

$$z_i = y_i^T R_k^{-1} y_i \tag{10}$$

where y_i is the pixel vector at site i and k is the bin into which y_i is classified. The z matrix can then be thresholded to provide detection. This transformation provides spectral decorrelation followed by energy detection. Use of this type of filter for anomaly detection has been proposed previously in [4] and [5].

Alternatively, it is possible to use the calculated bin centroids and cluster matrix as an initial state for a Bayesian classification algorithm. The goal of such an algorithm is to find the most likely classification of each pixel vector in terms of the MAP criterion. In order to do this, it is necessary to find the global maximum of (8). This is accomplished by using a modification of the adaptive clustering algorithm of Pappas [11], which was applied previously to single-band data in [12] and [13]. A flowchart for modified adaptive clustering, which is essentially a locally adaptive form of ICM [15], is given in Fig. 4.

Local adaptivity is achieved through the use of a shrinking estimation window of size W for parameter estimation. Each pixel in the image is assumed to have its own mean vector and covariance vector for each class. The mean vector $\mu_{(x,s)}$ is calculated based on all the pixels of class x within a window centered around site s, which is of size W. The corresponding variance vector is calculated similarly. Clearly, calculating mean and variance vectors for each class at each pixel in an image is a formidable computational burden. In order to ease this burden, the class means and variances are calculated at a grid of points spaced evenly throughout the image with a separation of $W/2$ pixels. Values at intermediate pixel sites are calculated through bilinear interpolation of the grid points. Initially, W is the size of the image, so only one mean and one variance vector are calculated by using all pixels in the image. When a stable solution is found at a given window size, W is halved and the process is repeated. This continues until some final value $W_{\min}$ is reached.

The core assumptions of this algorithm are that the data consists of a finite number of clusters in spectral space, each of which may be described by a multivariate Gaussian distribution, that class means and variances vary slowly with spatial position, and that pixels of similar composition will tend to cluster spatially as well as spectrally. Once each pixel has been classified and the local means and variances have been calculated for each class at each pixel site, (8) may be used to calculate the probability of each modeled pixel given the observed image data. Those pixels that are least probable, i.e., those which are the poorest matches for the image model, are most likely to be targets.

IV. Experimental Procedure and Results

Initial experiments with these algorithms were carried out on data obtained by using the Thermal Infrared Multispectral Sensor (TIMS) instrument. This camera measures six spectral bands ranging from 8.0 to 11.9 μm. Measurement is spectrally simultaneous (the images therefore do not require registration) and is taken a single line at a time from an airborne platform. Two TIMS data cubes were used in this study. The first was taken over an airfield in Kona, HI, and the second was

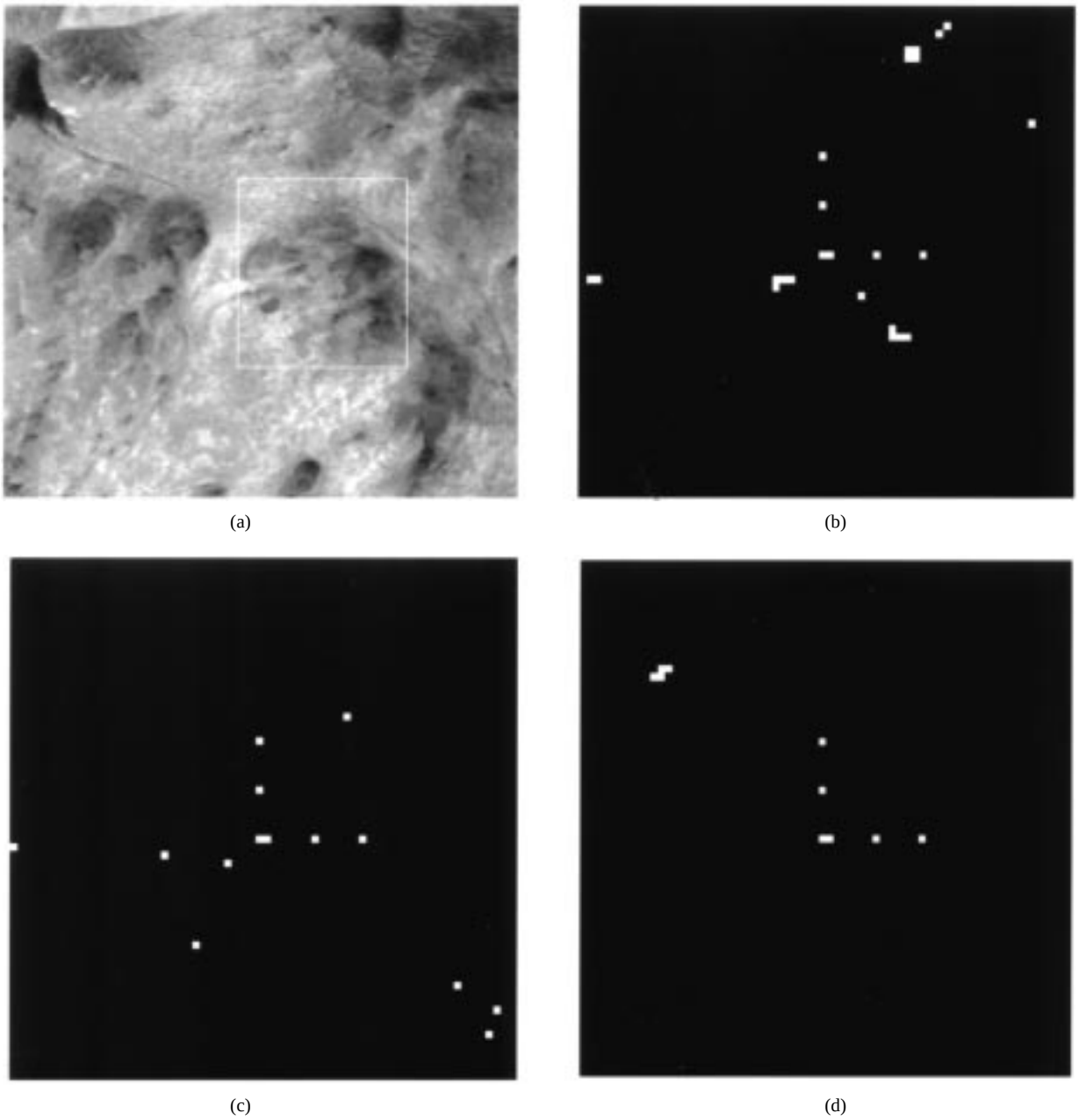


Fig. 7. (a) 200 × 200-pixel section from band 1 of the TIMS White Mountains image cube. Contains five targets implanted with a pixel-fill factor of 72%. (b) Smaller outlined section of (a), processed using RX filtering and thresholded at 100% detection. FAR is 0.56%. (c) Processed using spectral clustering followed by bin decorrelation. FAR is 0.19%. (d) Processed using Bayesian clustering. FAR is 0.10%.

taken over a region of the White Mountains in California. Neither of these image cubes contained known targets. Spectral measurements were taken of military vehicles by using the same sensor. The resultant spectral signatures were scaled to match the local background temperatures and inserted with 100% pixel fill into a number of background pixels in the image cubes. Linear interpolation between the resultant image cube and the original background cube was then used to produce image cubes with target pixel-fill factors ranging from 0 to 100% in 2% increments. This artificial insertion process was used because it allowed precise specification of target locations and pixel-fill factors, which made it possible to plot false-alarm rates (FAR) versus pixel-fill factors for both of the algorithms under consideration as well as for the spectral decorrelation method (the RX algorithm), which is currently being used in experimental small target-detection programs, such as the Coastal Battlefield Reconnaissance and Analysis (COBRA) system [19].

Fig. 5(a) shows a 200 × 200-pixel section from band 1 of the TIMS Kona image cube. This image contains ten targets inserted with a pixel-fill factor of 18%. Fig. 5(b)–(d) show the results of processing this image cube by applying the RX decorrelation filter, by modeling the background using spectral clustering and applying RX to the individual bins, and by modeling the background using Bayesian clustering. Each image is thresholded at a level that achieves 100% detection. Both clustering methods show improvement over filtering alone, with Bayesian clustering providing the best results.

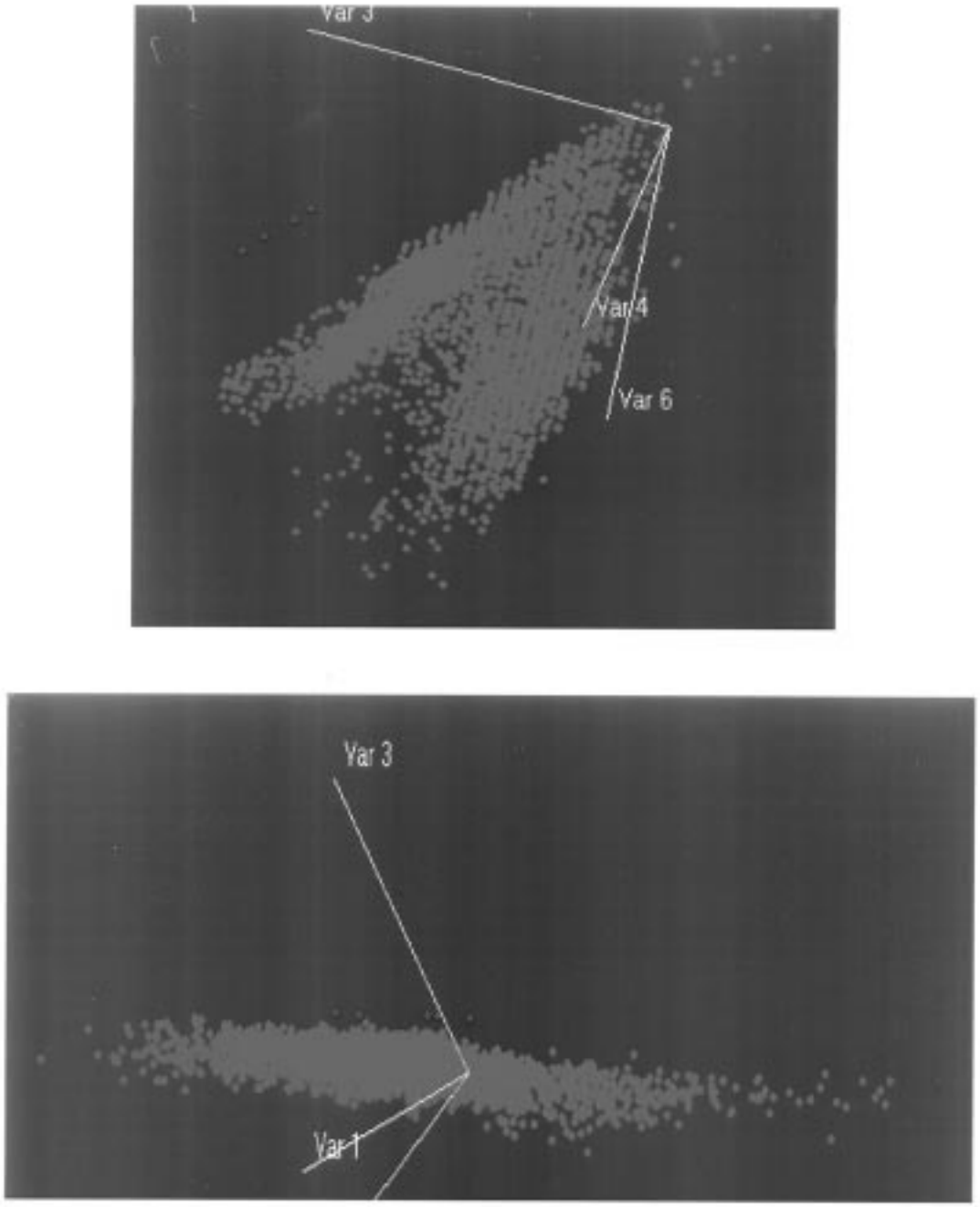
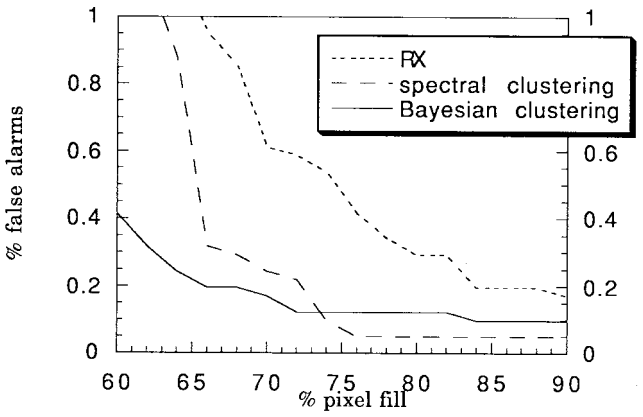


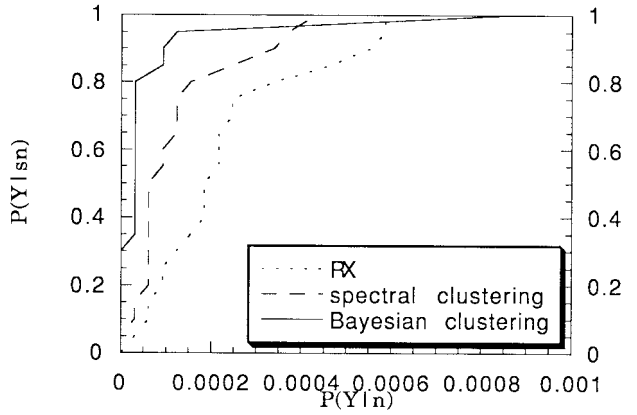
Fig. 8. Top: Scatterplot in three bands of targets (red) and background pixels (green) for the TIMS Kona data cube with 100% pixel-fill factor. Note that the targets are very distinct from the background. Gains from modeling are modest. Bottom: Scatterplot in three bands of target and background for the TIMS White Mountains data cube with 100% pixel-fill factor. Modeling produces larger gains due to the increased difficulty of the problem.

A thorough quantitative analysis of algorithm performance on this data set must plot the interactions of three variables: FAR, probability of detection, and percent pixel fill. In order to plot these variables in two dimensions, it is necessary to assign one a fixed value and examine the other two. Fig. 6(a) shows a plot of percent pixel fill versus FAR for the Kona data set, with probability of detection fixed at 100%. Note that all three algorithms fail catastrophically at pixel-fill factors of less than 14%. The reason for this is that, at this point, the target signatures are sufficiently mixed with the background pixels to cause them to be drawn into the background distribution. An alternate view of this data is given in Fig. 6(b), which shows the receiver operating characteristic (ROC) curves for all algorithms operating on the Kona data set, with pixel-fill factor fixed at 14%. Note that both plots show that the clustering algorithms provide significant improvement over RX filtering alone.

Fig. 7(a) shows a 200 $\times$ 200-pixel section from band 1 of the TIMS White Mountains image cube. This image contains five targets inserted with a pixel-fill factor of 72%. Fig. 7(b)–(d) show the results of processing this cube in the same manner as the Kona cube in Fig. 5. It should be noted in this case that first, the relative algorithm performances are consistent with those seen in the Kona data, i.e., that both



(a)



(b)

Fig. 9. (a) Plot of pixel fill versus FAR at 100% target detection for the TIMS White Mountains data set. (b) ROC curves at a pixel-fill factor of 72%. $P(Y|sn)$ denotes probability of detection, while $P(Y|n)$ denotes probability of false alarm. Note that the relative performances of the algorithms are consistent with those seen in the Kona data set, although the required fill factor for comparable performance is higher.

modeling algorithms provide improvements over RX filtering alone, second, that the amount of improvement is greater for both algorithms in this case than in the Kona data set, and third, that in this case a much higher pixel-fill factor is required to achieve similar levels of detection. The reasons for these observations are apparent upon examination of Fig. 8, which shows a scatterplot of background and target pixels for both data sets. In the Kona data, the targets are very spectrally distinct from the background. This is a relatively easy detection problem. RX performs adequately, and the gains from the modeling algorithms are modest. In the White Mountains data, however, the targets are very close to the background. This makes for a much more difficult detection problem and allows greater gains through use of the modeling algorithms. Fig. 9(a) shows a plot of percent pixel fill versus FAR for the White Mountains data set, with probability of detection fixed at 100%. Fig. 9(b) shows ROC curves for all algorithms operating on the White Mountains data set, with pixel fill fixed at 72%. The relative performance of the three algorithms in this case is consistent with what was seen for the Kona data, with the exception that, as expected, much higher values of pixel fill are required for equivalent levels of detection.

Once algorithm performance was quantified by using implanted targets, it was next necessary to ensure that similar performance could be obtained under realistic conditions in the field. Data for this set of experiments was obtained by using Night Vision Laboratory's linear variable filter (LVF) IR camera, which obtains 256 spectral bands ranging from 3 to 5 μm. Through elimination of redundant information and atmospheric absorption bands, this number can be reduced to 15 relevant spectral images. Because this instrument does not take images with spectral simultaneity, registration is an issue. This may be expected to degrade algorithm performance somewhat. The images under consideration were taken at night from high points overlooking the Shenandoah National Park near Luray, VA. Military vehicles were deployed in a realistic fashion along roadways in the Shenandoah Valley, at ranges sufficient to make them subpixel in extent. These image cubes are not very useful for quantitative study of algorithm performance because it is impossible to tell precisely what fraction of a pixel each target filled. However, as long as the locations of the targets are known, they are useful for relative comparisons.

Fig. 10(a) shows one band of an LVF data cube taken from the Hogback Overlook in Shenandoah National Park. There are two targets present in this image cube. Both are military vehicles painted camouflage, and both are at subpixel range. Fig. 10(b)–(d) shows the results of processing this cube. The results obtained in this case are consistent with those that were observed in the TIMS data with implanted targets. Both clustering algorithms provide marked improvement over the results obtained using RX alone. Similar results are seen in Fig. 11, which gives the results of processing an LVF data cube taken from the Mount Marshall Overlook. In this case, there are three targets, again all with subpixel extent. Although these experiments are less controlled than those involving the TIMS data, the reduction in false alarms that is obtained through the use of the clustering techniques appears to be reasonably consistent across all data sets.

V. Discussion

The results of the experiments described in the previous section appear to indicate significant utility for both clustering algorithms. In all cases, the number of false alarms is reduced without loss of sensitivity to the targets. However, there are two areas that these experiments do not address. The first is sensitivity to parameter variation. The spectral clustering algorithm has one variable parameter k, the number of classes. Vector quantization theory suggests that the algorithm's ability to model the background should approach optimality asymptotically as k increases. We would therefore expect the algorithm's performance to increase asymptotically toward some maximum as k increases. Fig. 12 shows a plot of probability of detection at a fixed FAR of 0.10% for $k = 1$ to $k = 24$ for both TIMS data sets. The Kona image cube in this case has a pixel-fill factor of 18%. The White Mountains image cube has a pixel-fill factor of 74%. In general, the algorithm's behavior is as expected. Probability of detection increases with k. However, both data sets display significant jitter. This occurs because, as k increases, the spectral means shift, moving closer to or farther away from the target-background mixtures in spectral space in a more or less random fashion.

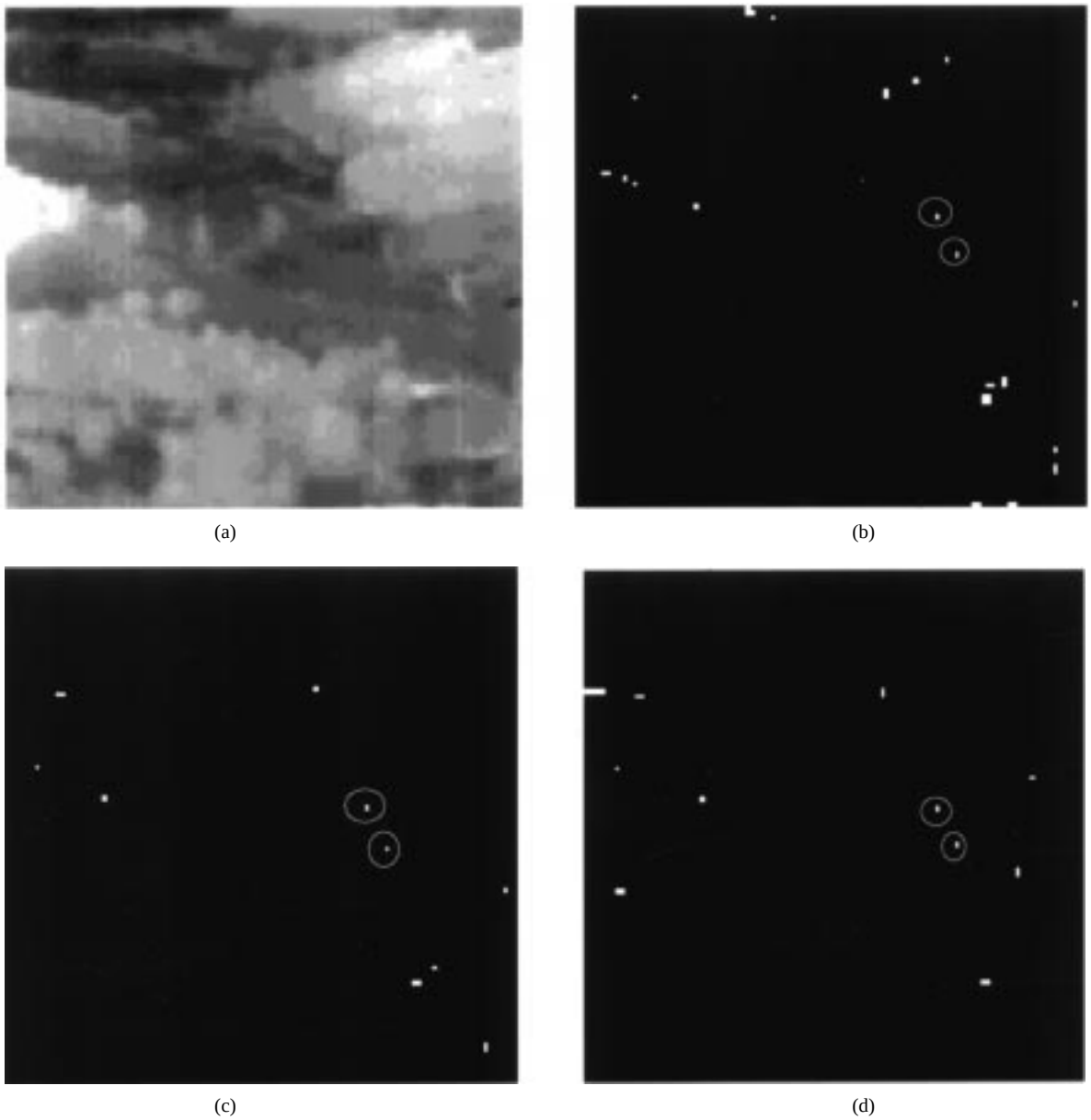
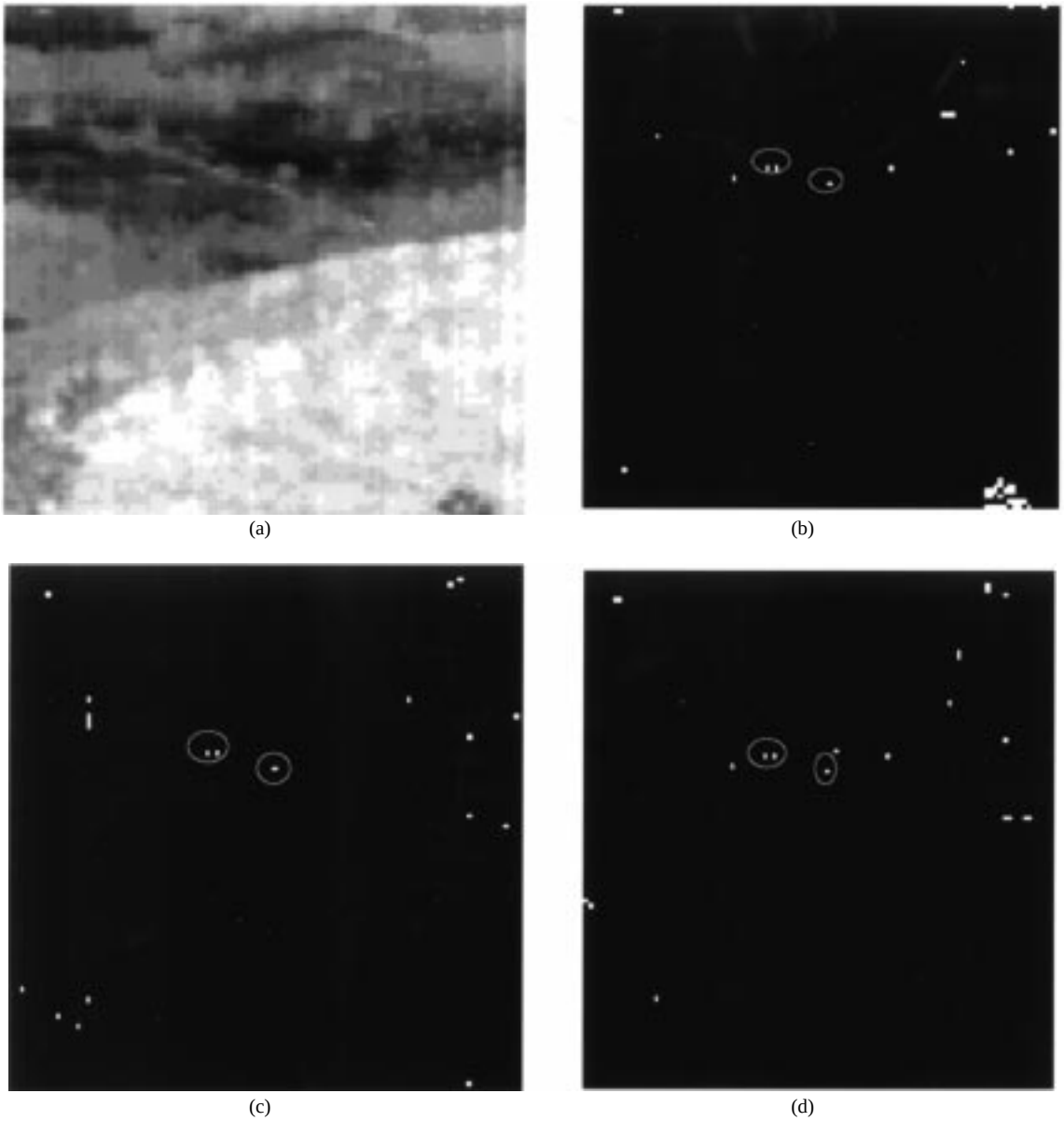


Fig. 10. (a) One band of the Hogback LVF data cube. Contains two targets that are of subpixel extent. (b) Processed using the RX filter followed by thresholding at the highest level, such that both targets were preserved. FAR is 0.33%. (c) Processed using spectral clustering followed by bin decorrelation. FAR is 0.11%. (d) Processed using Bayesian clustering. FAR is 0.16%.

These experiments indicate that it will be difficult to ensure optimality for this algorithm, particularly for the detection of targets with low pixel-fill factors. However, near-optimal performance appears to be attainable, so long as the value of k is kept relatively high. All experiments described in the previous section were carried out with $k = 24$.

The Bayesian clustering algorithm discussed here has three variable parameters: k, β, the weighting factor associated with the GRF, and $W_{\min}$, the minimum window size. Fortunately, previous work with GRF modeling [10]–[13], [20] suggests that this algorithm is quite stable, with regard to small variations in β. Although this paper describes a novel application of the model, experiments indicate that stability is present in this case as well. All experiments described in the previous section were carried out with $\beta = 1.0$. $W_{\min}$, which determines the extent of local adaptivity in the estimations of mean and variance, is a factor in algorithm performance only insofar as such local adaptivity is needed. In an image in which local means and variances change rapidly and by a large amount, a relatively small value for $W_{\min}$, on the order of 8–16, would be required. However, such conditions have not been observed in any of the data sets examined in this study. Our experiments show consistent results for values ranging from 8 to 64.

It can be expected that the Bayesian algorithm's sensitivity to k will be somewhat similar to that of the spectral clustering algorithm. Algorithm performance should increase with the number of classes. However, the random-field model should serve to prevent some of the jitter seen in Fig. 12 because, in order for a target pixel to be lost, it must be not only spectrally proximal to a class mean, but also spatially proximal to other pixels of that class. This observation is confirmed in Fig. 13, which plots algorithm performance versus k for



(a) (b) (c) (d)

Fig. 11. (a) One band from the Mount Marshall LVF data cube. Contains three targets that are of subpixel extent. (b) Processed using the RX filter followed by thresholding at a level such that all three targets were preserved. FAR is 0.40%. (c) Processed using spectral clustering followed by bin decorrelation. FAR is 0.17%. (d) Processed using Bayesian clustering. FAR is 0.19%. As in Fig. 10, the reduction in false alarms with the addition of both modeling algorithms is consistent with that seen in the TIMS data.

both TIMS data sets. Note that performance increases quickly to near-optimality and remains fairly stable thereafter. This property allows Bayesian classification to operate efficiently with a lower k value than the spectral clustering algorithm, and therefore saves a great deal of computation time. All experiments described in the previous section were carried out with $k = 6$.

The second issue not addressed in the previous section is speed of implementation. Clustering techniques, particularly Bayesian clustering techniques, which require stochastic relaxation solutions, have frequently been avoided in the past for applications involving multispectral imaging because of this issue. However, with certain minor modifications (such as the use of a causal neighborhood system), it is possible to implement both of the clustering algorithms described in this paper in real time. Our real-time processor makes use of a 10 000 pixel training sequence for initial estimates of class means and variances. This process requires approximately 1 min of processing time on a Silicon Graphics Indigo workstation. Once the initial estimates are obtained, the Bayesian algorithm is able to process a steady data stream at a rate of approximately 500 pixels/s, including window management time. In this sort of implementation, it is of course impossible to make use of an iteratively shrinking estimation window, but means and variances are updated on a line-by-line basis, so some measure of local adaptivity is retained. Initial experiments have shown that the drop off in performance required by real-time implementation is minor. Initialization for the spectral clustering algorithm requires approximately 4 min, due to the higher value of k. Sensitivity of this algorithm's real-time implementation is virtually identical to that described in the previous section.

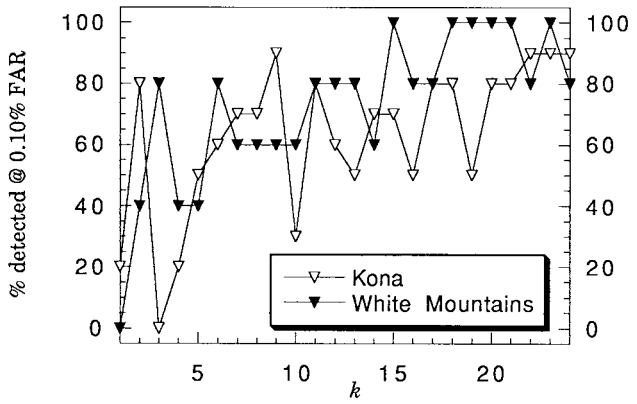


Fig. 12. Plot of probability of detection at a FAR of 0.10% versus k (number of bins) for the Kona TIMS data with a pixel-fill factor of 18% and the White Mountains TIMS data with a pixel-fill factor of 74%. Detection algorithm is spectral clustering followed by bin decorrelation. Although algorithm performance does increase with k, there is considerable irregularity in both data sets. Near-optimality may only be assured by making use of a large value for k.

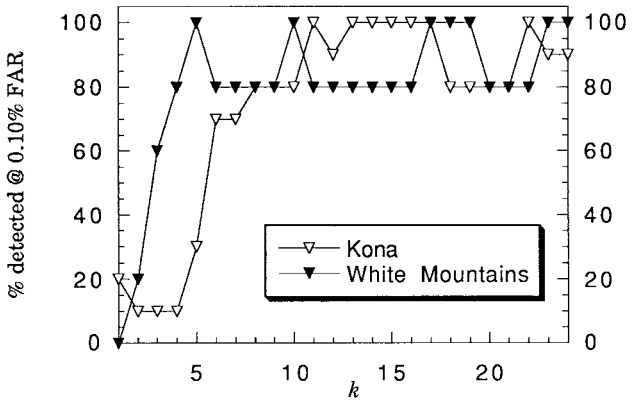


Fig. 13. Plot of probability of detection at a FAR of 0.10% versus k for the Kona TIMS data with a pixel-fill factor of 18% and the White Mountains TIMS data with a pixel-fill factor of 74%. Detection algorithm is Bayesian classification. As in Fig. 12, probability of detection increases with k. However, this algorithm shows much more stability. Near-optimality may therefore be achieved with lower values of k, allowing improved speed of implementation.

One final result from the experiments described in the previous section bears mentioning. It may already have been noted during examination of Figs. 5, 7, 10, and 11 that the false alarms produced by the three algorithms examined here are only very weakly spatially correlated. It may therefore be possible to achieve results that are superior to any of those presented here through parallel processing of data using multiple algorithms followed by data fusion using a logical AND. This is one topic for future examination.

The generally consistent results obtained for different data sets and sensor types using both of the novel algorithms described in this paper indicate that either might be useful as a cuing processor for long-range surveillance applications. However, the Bayesian algorithm's higher implementation speed and greater stability, with respect to parameter variations, indicate that it probably provides the best solution to this problem. Future work in this area will include code optimization for both algorithms and verification of their efficacy in the field.

REFERENCES

[1] 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, May 1985.

[2] L. Hoff, J. Zeidler, and C. Yerkes, "Adaptive multispectral image processing for the detection of small targets in terrain clutter," *SPIE Signal Data Processing Small Targets*, vol. 1698, pp. 100–114, Apr. 1992,.

[3] M. Schmalz and G. Ritter, "Image-algebraic design of multispectral target recognition algorithms," *Image Algebra and Morphological Image Processing V*, vol. 2300. Washington, DC: SPIE, July 1994, pp. 213–228.

[4] T. Soni, J. Zeidler, and W. H. Ku, "Adaptive whitening filters for small target detection," *SPIE Signal Data Processing Small Targets*, vol. 1698, pp. 21–31, Apr. 1992,.

[5] 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, pp. 1760–1770, Oct. 1990.

[6] X. Yu, I. S. Reed, and A. Stocker, "Comparative performance analysis of adaptive multispectral detectors," *IEEE Trans. Signal Processing*, vol. 41, pp. 2639–2656, Aug. 1993.

[7] E. Rignot and R. Chellappa, "Algorithms for segmentation of complex-amplitude SAR data," *NASA Tech Briefs*, vol. 17, pp. 28–30, June 1993.

[8] Y. Linde, A. Buzo, and R. Gray, "An algorithm for vector quantizer design," *IEEE Trans. Commun. Theory*, vol. COM-28, pp. 84–95, Jan. 1980.

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

[10] H. Derin and H. Elliott, "Modeling and segmentation of noisy and textured images using Gibbs random field," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. PAMI-9, pp. 39–55, Jan. 1987.

[11] T. Pappas, "An adaptive clustering algorithm for image segmentation," *IEEE Trans. Signal Processing*, vol. 40, pp. 901–914, Apr. 1992.

[12] E. A. Ashton, M. Berg, K. Parker, J. Weisberg, C. Chen, and L. Ketonen, "Segmentation and feature extraction techniques, with applications to MRI head studies," *Magnetic Resonance Med.*, vol. 33, pp. 670–677, Mar. 1995.

[13] E. Ashton and K. Parker, "Multiple resolution Bayesian segmentation of ultrasound images," *Ultrasonic Imaging*, vol. 17, pp. 291–304, Oct. 1995.

[14] T. Watanabe, H. Suzuki, S. Tanba, and R. Yokoyama, "Improved contextual classifiers of multispectral image data," *IEICE Trans. Fundamentals Elec., Commun., Comput. Sci.*, vol. E77-A, pp. 1445–1450, Sept. 1994.

[15] J. Besag, "On the statistical analysis of dirty pictures," *J. R. Stat. Soc.*, vol. 48, pp. 259–302, Mar. 1986.

[16] R. Dubes and A. Jain, "Random field models in image analysis," *J. Appl. Stat.*, vol. 16, pp. 131–164, Feb. 1989.

[17] C. Bouman and B. Liu, "Multiple resolution segmentation of textured images," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 13, pp. 99–113, Feb. 1991.

[18] R. Gonzales and R. Woods, *Digital Image Processing.* New York: Addison-Wesley, June 1992.

[19] R. Muise, "Coastal mine detection using the COBRA multispectral sensor," *SPIE Detection Remediation Technol. Mines Minelike Targets*, vol. 2765, pp. 15–24, Apr. 1996,.

[20] E. A. Ashton, "Segmentation and feature extraction techniques, with applications to biomedical images," Ph.D. dissertation, Univ. of Rochester, Rochester, NY, Aug. 1995.

Edward A. Ashton received the B.S.E.E. degree from Loyola College, Baltimore, MD, in 1991 and the M.S.E.E. and Ph.D. degrees in electrical engineering from the University of Rochester, Rochester, NY, in 1993 and 1995, respectively.

He began working for the Naval Research Laboratory, Washington, DC, in 1995. His research interests include multi- and hyperspectral-image exploitation, machine vision, automated image analysis, and deformable model theory.