JOINT SPARSITY MODEL FOR TARGET DETECTION IN HYPERSPECTRAL IMAGERY

Yi Chen
The Johns Hopkins University
3400 N. Charles Street, Baltimore, MD 21218

Nasser M. Nasrabadi
US Army Research Laboratory
2800 Powder Mill Road, Adelphi, MD 20783

Trac D. Tran*
The Johns Hopkins University
3400 N. Charles Street, Baltimore, MD 21218

ABSTRACT

This letter proposes a simultaneous joint sparsity model for target detection in hyperspectral imagery. The key innovative idea here is that hyperspectral pixels within a small neighborhood in the test image are simultaneously represented by a linear combination of a few common training samples but weighted with a different set of coefficients for each pixel. The joint sparsity model automatically incorporates the inter-pixel correlation within the hyperspectral imagery by assuming that neighboring pixels usually consists of similar materials. The sparse representations of the neighboring pixels are obtained by simultaneously decomposing the pixels over a given dictionary consisting of training samples of both the target and background classes. The recovered sparse coefficient vectors are then directly used for determining the label of the test pixels. Simulation results show that the proposed algorithm outperforms the classical hyperspectral target detection algorithms, such as the popular spectral matched filters, matched subspace detectors, adaptive subspace detectors, as well as binary classifiers such as support vector machines.

1. INTRODUCTION

Most natural signals are inherently sparse in certain basis or with respect to a given dictionary. They can be approximately represented by a few coefficients carrying the most relevant information. The sparsity of signals has played a very important role in many classical signal processing applications such as compression and image denoising. The recent development in sparse modeling of signals and images (Bruckstein et al. 2009) has provided an extremely powerful tool for computer vision and pattern recognition (Wright et al. 2010).

In hyperspectral imagery (HSI), pixels are represented by vectors whose entries correspond to spectral bands, and images are represented by three-dimensional cubes. One of the most important applications of HSI is target detection, which can be viewed as a binary classification problem where pixels are labeled as target or background based on their spectral characteristics. Support vector machines (SVM) have been a powerful tool to solve supervised classification problems and have shown good performances in hyperspectral classification (Melgani and Bruzzone 2004; Camps-Valls et al. 2006). A number of statistical hypothesis testing techniques have also been proposed for target detection in HSI (Manolakis and Shaw 2002). Among these approaches, spectral matched filter (SMF), matched subspace detectors (MSD), and adaptive subspace detectors (ASD) have been widely used to detect various targets of interests (Kwon and Nasrabadi 2007).

In this letter, we propose a new HSI target detection algorithm based on a joint sparsity model (Tropp et al. 2006; Cotter et al. 2005) for pixels in a small neighborhood. It is observed that pixels belonging to the same class approximately lie in a low-dimensional subspace. Therefore, an unknown pixel lies in the union of the low-dimensional target and background subspaces and can be approximately represented by very few training samples from target or background sub-dictionaries. However, a direct application of this pixel-wise sparse representation would ignore spatial information in the detection process. In this letter, we incorporate the spatial information from neighboring pixels by using the joint sparsity model where pixels in a small neighborhood are assumed to be simultaneously represented by a linear combination of a few common training samples, but for each pixel these training samples are weighted with a different set of coefficients. In this way, we force the representations of neighboring pixels to have a common sparse

support with respect to the training dictionary. The support is recovered by simultaneously decomposing the test pixels over the training dictionary. The recovery process implicitly involves a competition between the target and background subspaces and the recovered sparse representation is naturally discriminative. The labels of the test samples are then directly determined by the property of the recovered sparse vectors.

The letter is structured as follows. The joint sparsity model and the proposed target detection algorithm is presented in Section 2. The effectiveness of the proposed method is demonstrated by simulation results presented in Section 3. Conclusions are drawn in Section 4.

2. SIMULTANEOUS JOINT SPARSITY MODEL FOR HSI TARGET DETECTION

In this section, we first briefly introduce the HSI target detection technique based on sparse representation for a single pixel. Next, we show how to incorporate a joint sparsity constraint across neighboring pixels of HSI by adopting the simultaneous joint sparsity model (Cotter et al. 2005; Tropp et al. 2006).

1. Pixel-wise Sparsity-based Target Detection

Let $\boldsymbol{x} \in \mathbb{R}^B$ be a B-dimensional hyperspectral pixel observation whose entries correspond to the spectral bands. The spectrum of $\boldsymbol{x}$ is modeled to lie in the union of two low-dimensional subspaces: the background and target subspaces spanned by background training samples $\{\boldsymbol{a}_i^b\}_{i=1,2,\ldots,N_b}$ and target training samples $\{\boldsymbol{a}_i^t\}_{i=1,2,\ldots,N_t}$, respectively. Therefore, $\boldsymbol{x}$ can be written as a sparse linear combination of all training pixels

$$\begin{aligned}\boldsymbol{x} &= \left(\alpha_1^b \boldsymbol{a}_1^b + \cdots + \alpha_{N_b}^b \boldsymbol{a}_{N_b}^b\right) + \left(\alpha_1^t \boldsymbol{a}_1^t + \cdots + \alpha_{N_t}^t \boldsymbol{a}_{N_t}^t\right) \\ &= \underbrace{\begin{bmatrix}\boldsymbol{a}_1^b & \cdots & \boldsymbol{a}_{N_b}^b\end{bmatrix}}_{\boldsymbol{A}^b} \underbrace{\begin{bmatrix}\alpha_1^b \\ \vdots \\ \alpha_{N_b}^b\end{bmatrix}}_{\boldsymbol{\alpha}^b} + \underbrace{\begin{bmatrix}\boldsymbol{a}_1^t & \cdots & \boldsymbol{a}_{N_t}^t\end{bmatrix}}_{\boldsymbol{A}^t} \underbrace{\begin{bmatrix}\alpha_1^t \\ \vdots \\ \alpha_{N_t}^t\end{bmatrix}}_{\boldsymbol{\alpha}^t} \\ &= \boldsymbol{A}^b\boldsymbol{\alpha}^b + \boldsymbol{A}^t\boldsymbol{\alpha}^t = \underbrace{\begin{bmatrix}\boldsymbol{A}^b & \boldsymbol{A}^t\end{bmatrix}}_{\boldsymbol{A}} \underbrace{\begin{bmatrix}\boldsymbol{\alpha}^b \\ \boldsymbol{\alpha}^t\end{bmatrix}}_{\boldsymbol{\alpha}} = \boldsymbol{A}\boldsymbol{\alpha}.\end{aligned} \quad (1)$$

In the above equation, $\boldsymbol{A}^b$ and $\boldsymbol{A}^t$ are the background and target sub-dictionaries consisting of the N_b background and N_t target training samples (also called atoms), respectively, whereas $\boldsymbol{\alpha}^b$ and $\boldsymbol{\alpha}^t$ are vectors whose entries correspond to the atoms in $\boldsymbol{A}^b$ and $\boldsymbol{A}^t$, respectively. The matrix $\boldsymbol{A} \in \mathbb{R}^{B\times N}$ with $N = N_b + N_t$ is the training dictionary consisting of both background and target training samples and $\boldsymbol{\alpha} \in \mathbb{R}^N$ is a concatenation of the two vectors $\boldsymbol{\alpha}^b$ and $\boldsymbol{\alpha}^t$. In this sparsity model, $\boldsymbol{\alpha}$ turns out to be a sparse vector (i.e., a vector with only few non-zero entries). The number K of nonzero entries in $\boldsymbol{\alpha}$ is called the sparsity level of $\boldsymbol{\alpha}$ and the index set Λ_K on which the entries of $\boldsymbol{\alpha}$ are nonzero is called the support of $\boldsymbol{\alpha}$. The sparse representation $\boldsymbol{\alpha}$ is very discriminative and contains important information about the class of the test sample $\boldsymbol{x}$.

Given the training dictionary $\boldsymbol{A}$, the sparse representation $\boldsymbol{\alpha}$ satisfying $\boldsymbol{A}\boldsymbol{\alpha} = \boldsymbol{x}$ can be obtained by solving the following problem:

$$\hat{\boldsymbol{\alpha}} = \arg\min \|\boldsymbol{\alpha}\|_0 \quad \text{subject to} \quad \boldsymbol{A}\boldsymbol{\alpha} = \boldsymbol{x}, \quad (2)$$

where $\|\cdot\|_0$ denotes ℓ_0-norm which is defined as the number of non-zero entries in the vector. The above problem is NP-hard, which can be relaxed to a linear programming, if the solution is sufficiently sparse, by replacing the ℓ_0-norm by ℓ_1-norm and solved by convex programming techniques (Bruckstein et al. 2009). The problem in (2) can also be solved by greedy pursuit algorithms such as Orthogonal Matching Pursuit (OMP) (Tropp and Gilbert 2007) which efficiently approximates the solution with computational complexity $\mathcal{O}(BNK)$ for K iterations. Due to the presence of approximation error in empirical data, the equality constraint can be relaxed to an inequality one

$$\hat{\boldsymbol{\alpha}} = \arg\min \|\boldsymbol{\alpha}\|_0 \quad \text{subject to} \quad \|\boldsymbol{A}\boldsymbol{\alpha} - \boldsymbol{x}\|_2 \le \sigma, \quad (3)$$

where σ is the error tolerance. The above problem can also be interpreted as minimizing the approximation error with certain sparsity level

$$\hat{\boldsymbol{\alpha}} = \arg\min \|\boldsymbol{A}\boldsymbol{\alpha} - \boldsymbol{x}\|_2 \quad \text{subject to} \quad \|\boldsymbol{\alpha}\|_0 \le K_0, \quad (4)$$

where K_0 is a given upper bound on the sparsity level (Tropp and Wright 2010). In fact, the greedy algorithm OMP solves (3) or (4) depending on the stopping criterion of the algorithm, which will be explained in more details in the next section.

The decomposition of the test sample $\boldsymbol{x}$ over the entire training dictionary $\boldsymbol{A}$ for the few most representative atoms leads to a competition between the two subspaces and therefore the recovered sparse vector $\boldsymbol{\alpha}$ is itself discriminative. The class of $\boldsymbol{x}$ can be determined by comparing the residuals $r_b(\boldsymbol{x}) = \left\|\boldsymbol{x} - \boldsymbol{A}^b\hat{\boldsymbol{\alpha}}^b\right\|_2$ and $r_t(\boldsymbol{x}) = \|\boldsymbol{x} - \boldsymbol{A}^t\hat{\boldsymbol{\alpha}}^t\|_2$, where $\hat{\boldsymbol{\alpha}}^b$ and $\hat{\boldsymbol{\alpha}}^t$ represent the recovered sparse coefficients corresponding to the background and target sub-dictionaries, respectively. The output of detector is calculated by

$$D(\boldsymbol{x}) = r_b(\boldsymbol{x}) - r_t(\boldsymbol{x}) = \left\|\boldsymbol{x} - \boldsymbol{A}^b\hat{\boldsymbol{\alpha}}^b\right\|_2 - \left\|\boldsymbol{x} - \boldsymbol{A}^t\hat{\boldsymbol{\alpha}}^t\right\|_2. \quad (5)$$

If $D(\boldsymbol{x}) > \delta$ with δ being a prescribed threshold, then $\boldsymbol{x}$ is determined as a target pixel; otherwise, $\boldsymbol{x}$ is labeled as a background pixel.

2. Joint Sparsity Model

In the previous section, detection is performed for each pixel in the test image independently regardless of the correlation between neighboring pixels. However, in a typical

HSI, neighboring pixels usually consist of similar materials and thus their spectral characteristics are highly correlated. To exploit this crucial inter-pixel correlation, we could explicitly include an additional smoothing term, such as the Laplacian (Chen et al. 2010) or total variation (Rudin et al. 1992), in the optimization problem formulation (2). However, in this paper we propose to exploit the inter-pixel correlation by adopting the joint sparsity model, in which the underlying sparse representations of multiple neighboring pixels share a common sparsity pattern. Since neighboring HSI pixels consist of similar materials, they can be approximated by a sparse linear combination of a few common atoms in the training dictionary but weighted with a different set of coefficients. The problem of simultaneous sparse representation and approximation for multiple measurement vectors has previously been studied in the literature (Cotter et al. 2005; Tropp et al. 2006), but it has never been applied in HSI for target detection and classification.

The joint sparsity model for pixels in a small neighborhood $\mathscr{N}_\varepsilon$ consisting of T pixels is formulated as follows. Let $\boldsymbol{A}$ be the $B\times N$ training dictionary and $\boldsymbol{X}=\begin{bmatrix}\boldsymbol{x}_1 & \boldsymbol{x}_2 & \cdots & \boldsymbol{x}_T\end{bmatrix}$ be a $B\times T$ data matrix whose columns $\{\boldsymbol{x}_t\}_{t=1,\ldots,T}$ are pixels in $\mathscr{N}_\varepsilon$. Since pixels in $\mathscr{N}_\varepsilon$ could consist of similar materials, they should be represented by a linear combination of a common set of K training samples $\{\boldsymbol{a}_{\lambda_1},\ldots,\boldsymbol{a}_{\lambda_K}\}$. Specifically, the sparse representation of $\boldsymbol{x}_t\in\mathscr{N}_\varepsilon$ is expressed as

$$\boldsymbol{x}_t=\boldsymbol{A}\boldsymbol{\alpha}_t=\alpha_{t,\lambda_1}\boldsymbol{a}_{\lambda_1}+\alpha_{t,\lambda_2}\boldsymbol{a}_{\lambda_2}+\cdots+\alpha_{t,\lambda_K}\boldsymbol{a}_{\lambda_K}.$$

The index set $\Lambda_K=\{\lambda_1,\lambda_2,\ldots,\lambda_K\}$ is the support of $\boldsymbol{\alpha}_t$ and each pixel $\boldsymbol{x}_t$ has a different set of coefficients $\{\alpha_{t,k}\}_{k\in\Lambda_K}$ associated with it. The data matrix $\boldsymbol{X}$ can then be represented as

$$\boldsymbol{X}=\begin{bmatrix}\boldsymbol{x}_1 & \boldsymbol{x}_2 & \cdots & \boldsymbol{x}_T\end{bmatrix}=\boldsymbol{A}\underbrace{\begin{bmatrix}\boldsymbol{\alpha}_1 & \boldsymbol{\alpha}_2 & \cdots & \boldsymbol{\alpha}_T\end{bmatrix}}_{\boldsymbol{S}}=\boldsymbol{A}\boldsymbol{S}, \tag{6}$$

where each of the sparse vectors $\{\boldsymbol{\alpha}_t\}_{t=1,\ldots,T}$ has the same support Λ_K and $\boldsymbol{S}$ is a sparse matrix with only K nonzero rows. For convenience, we call the index set Λ_K of $\boldsymbol{\alpha}_t$ also the support of $\boldsymbol{S}$.

Given the dictionary $\boldsymbol{A}$, the matrix $\boldsymbol{S}$ is obtained by solving the following joint sparse recovery problem.

$$\begin{aligned}\text{minimize}\quad & \|\boldsymbol{S}\|_{\text{row},0}\\ \text{subject to:}\quad & \boldsymbol{A}\boldsymbol{S}=\boldsymbol{X},\end{aligned} \tag{7}$$

where the notation $\|\boldsymbol{S}\|_{\text{row},0}$ denotes the number of non-zero rows of $\boldsymbol{S}$ (also called the diversity of $\boldsymbol{S}$ in (Cotter et al. 2005)). The solution to the above problem

$$\hat{\boldsymbol{S}}=\begin{bmatrix}\hat{\boldsymbol{\alpha}}_1 & \hat{\boldsymbol{\alpha}}_2 & \cdots & \hat{\boldsymbol{\alpha}}_T\end{bmatrix} \tag{8}$$

is an $N\times T$ sparse matrix with only few nonzero rows. Similar to the sparse recovery problem in (2), the simultaneous sparse recovery problem in (7) is an NP-hard problem. It can be solved by greedy algorithms (Cotter et al. 2005; Tropp et al. 2006), or relaxed to a convex programming (Bruckstein et al. 2009). For empirical data, the problem in (7) can also be rewritten to account for the approximation errors (Rakotomamonjy submitted for publication, 2010), as it was done in (3) and (4):

$$\begin{aligned}\text{minimize}\quad & \|\boldsymbol{S}\|_{\text{row},0}\\ \text{subject to:}\quad & \|\boldsymbol{A}\boldsymbol{S}-\boldsymbol{X}\|_F\le\sigma,\end{aligned} \tag{9}$$

or

$$\begin{aligned}\text{minimize}\quad & \|\boldsymbol{A}\boldsymbol{S}-\boldsymbol{X}\|_F\\ \text{subject to:}\quad & \|\boldsymbol{S}\|_{\text{row},0}\le K_0.\end{aligned} \tag{10}$$

In this letter, the simultaneous sparse recovery problem is solved by a greedy algorithm, called the Simultaneous Orthogonal Matching Pursuit (SOMP), whose implementation details are presented in (Tropp et al. 2006). SOMP recovers the common support set Γ_K in a sequential fashion (i.e., atoms in the dictionary $\boldsymbol{A}$ is sequentially selected). At each iteration, the atom that simultaneously yields the best approximation to all of the residual vectors is selected. The SOMP algorithm terminates when the error residual $\|\boldsymbol{A}\boldsymbol{S}-\boldsymbol{X}\|_F$ is sufficiently small, or the desired level of sparsity (controlled by the number of iterations) is achieved, which solves the problems in (9) or (10), respectively.

After the sparse matrix $\hat{\boldsymbol{S}}$ is recovered, the labels of the test samples can be determined based on the characteristics of the sparse coefficients as is done in Section 1. We calculate and compare the total error residuals between the original test samples and the approximations obtained from the background and target sub-dictionaries. The output of the proposed sparsity-based detector is computed as in (5) by the difference of the total residuals from all of the pixels in the neighborhood

$$D(\boldsymbol{x})=\left\|\boldsymbol{X}-\boldsymbol{A}^b\hat{\boldsymbol{S}}^b\right\|_F-\left\|\boldsymbol{X}-\boldsymbol{A}^t\hat{\boldsymbol{S}}^t\right\|_F, \tag{11}$$

where $\hat{\boldsymbol{S}}^b$ consists of the first N_b rows of the recovered matrix $\hat{\boldsymbol{S}}$ corresponding to the background sub-dictionary $\boldsymbol{A}^b$, and $\hat{\boldsymbol{S}}^t$ consists of the remaining N_t rows in $\hat{\boldsymbol{S}}$ corresponding to the target sub-dictionary $\boldsymbol{A}^t$. If the output is greater than a prescribed threshold, then the test sample is labeled as a target; otherwise it is labeled as background.

An example of the proposed detection technique based on the joint sparsity model is illustrated in Fig. 1. The test pixel $\boldsymbol{x}_1$ is taken from the background region in the upper-right corner of the Desert Radiance II data collection, shown in Fig. 3(a). More details about this image will be discussed in Section 3. The dictionary $\boldsymbol{A}$ consists of $N_t=18$ target training samples and $N_b=216$ background training samples. Using the sparsity model for a single pixel in (1) and the OMP algorithm, the sparse vector $\hat{\boldsymbol{\alpha}}_1$ of this background involves both background and target training samples, as seen

in Fig. 1(a). Neither of the approximations obtained from the background and target sub-dictionaries represents well the original test pixel (Fig. 1(b)). On the other hand, when incorporating the spatial correlation by applying the joint sparsity model in (6) on a 3×3 neighborhood centered at $\boldsymbol{x}_1$, the test sample $\boldsymbol{x}_1$ is well approximated by the background sub-dictionary, as seen in Fig. 1(d). Therefore, with the appropriate help from the joint sparsity model, the test pixel is more accurately represented as background.

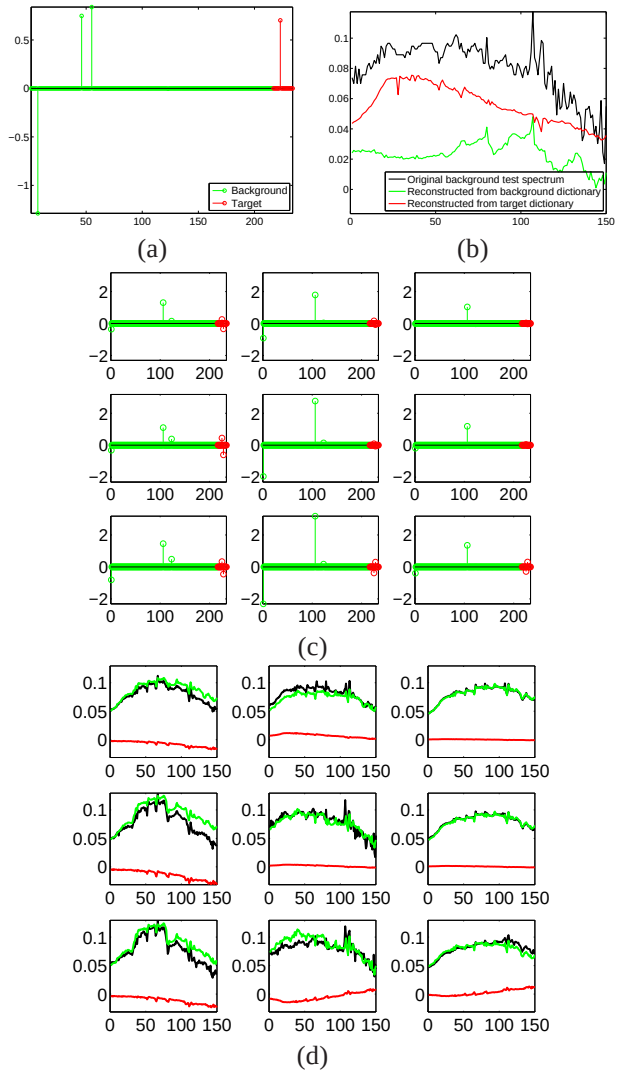


Fig. 1. An example comparing the pixel-wise sparsity model and the joint sparsity model for a background test sample $\boldsymbol{x}_1$. (a) Solution to (4) (pixel-wise sparsity model). (b) From (4), the reconstructions from the background sub-dictionary (green) and from the target sub-dictionary (red), and the original test sample (black). (c) Solution to (10) (joint-sparsity model) for a 3×3 window centered at the pixel $\boldsymbol{x}_1$. (d) From (10), the reconstructions from the background sub-dictionary (green) and from the target sub-dictionary (red), and the original test sample (black).

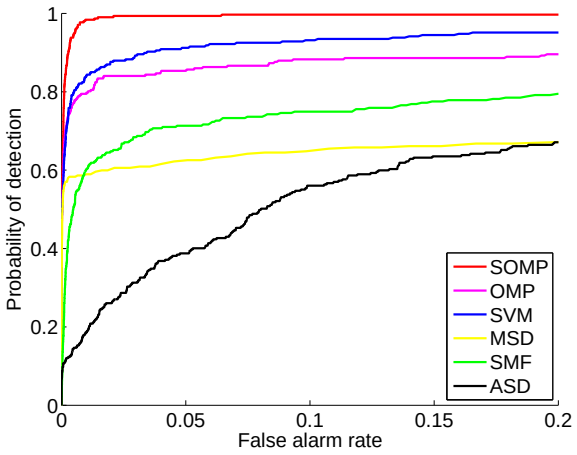
3. SIMULATION RESULTS AND ANALYSIS

The proposed target detection algorithm, as well as the classical techniques SMF, MSD, ASD, and SVM are applied to several real HSI. The results are compared both visually and quantitatively by the receiver operating characteristics (ROC) curves. The ROC curve describes the probability of detection (PD) as a function of the probability of false alarms (PFA). To calculate the ROC curve, we pick thousands of thresholds between the minimum and maximum of the detector output. The class labels for all pixels in the test region are determined at each threshold. The PFA is calculated by the number of false alarms (background pixels determined as target) over the total number of pixels in the test region, and the PD is the ratio of the number of hits (target pixels determined as target) and the total number of true target pixels.

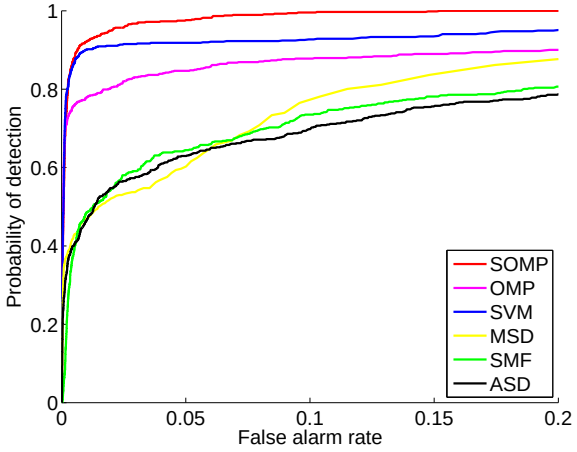
The two hyperspectral images used in the experiments, the desert radiance II data collection (DR-II) and forest radiance I data collection (FR-I), are from the hyperspectral digital imagery collection experiment (HYDICE) sensor (Basedow et al. 1995). The HYDICE sensor generates 210 bands across the whole spectral range from 0.4 to 2.5 μm which includes the visible and short-wave infrared bands. We use 150 of the 210 bands (23rd-101st, 109th-136th, and 152nd-194th) by removing the absorption and low-SNR bands. The DR-II image contains 6 military targets on the dirt road, while the FR-I image contains 14 targets along the tree line, as seen in Fig. 3(a) and Fig. 4(a), respectively. For these two HYDICE images, every pixel on the targets is considered a target pixel when computing the ROC curves. We use a small target sub-dictionary constructed by $N_t = 18$ pixels on the leftmost target in the scene. For the background sub-dictionary, we use an adaptive local background dictionary. Specifically, the background sub-dictionary $\boldsymbol{A}^b$ is generated locally for each test pixel using a dual window centered at the pixel of interest. The inner window should be larger than the size of the targets in the scene and only pixels in the outer region will form the atoms in $\boldsymbol{A}^b$. In this way, the subspace spanned by the background sub-dictionary becomes adaptive to the local statistics. For DR-II and FR-I, the outer and inner windows have size 21×21 and 15×15, respectively, so there are $N_b = 216$ background training samples in the dictionary.

The detection techniques using the single-pixel sparsity model (1) and the joint sparsity model (6) are applied on these two HYDICE images. The OMP algorithm (Tropp and Gilbert 2007) is used to solve the sparsity-constrained problem in (4). The SOMP algorithm is used to solve the simultaneous sparse recovery problem in (10). The ROC curves obtained using OMP and SOMP are shown in Fig. 2, with the sparsity level being set to $K_0 = 10$. For SOMP, a 5×5 square neighborhood ($T = 25$ in the problem formulation (6)) is used. The detector output for these two sparsity-based algorithms are shown in Figs. 3(c)-(d) for DR-II and

Figs. 4(c)-(d) for FR-I. We see that the incorporation of the spatial correlation between neighboring pixels through the joint sparsity model yields better detection results than a direct application of the pixel-wise sparsity model (1).



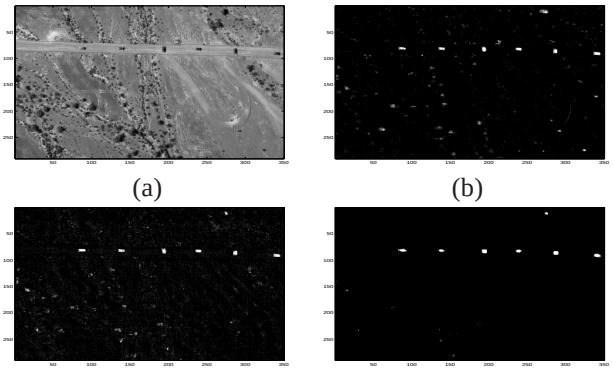
(a)



(b)

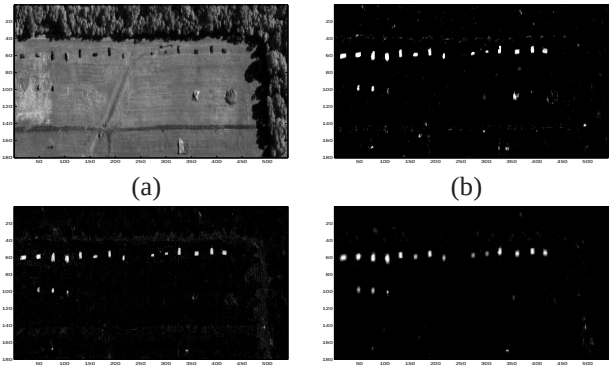
Fig. 2. ROC curves using various detection and classification algorithms for (a) DR-II and (b) FR-I using local background dictionary (dual window approach), $N_t = 18$ and $N_b = 216$.

Under the same settings (i.e., same target and background training samples for all detectors), the classical statistical detectors SMF, MSD, and ASD as well as the binary classifier SVM are also applied to detect the targets of interests. The first three statistical detectors are pixel-wise detectors and their implementation details can be found in (Kwon and Nasrabadi 2007). For SVM, we use a composite kernel that combines the spectral and spatial information via a weighted summation, which is shown to outperform the spectral-only SVM in HSI classification (Camps-Valls et al. 2006). For each pixel, the spatial features (e.g., the mean and the standard deviation per spectral band) are explicitly extracted in the 5×5 neighborhood centered at that pixel. A SVM is then trained for each spectral-spatial pixel using atoms in $\boldsymbol{A}^b$ and $\boldsymbol{A}^t$ as belonging to two different classes with RBF kernels for



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

Fig. 3. DR-II. (a) Averaged image over 150 bands. Detection results for DR-II using (b) SVM with composite kernel, (c) OMP, (d) SOMP.



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

Fig. 4. FR-I. (a) Averaged image over 150 bands. Detection results for FR-I using (b) SVM with composite kernel, (c) OMP, (d) SOMP.

both spectral and spatial features. The parameters associated with SVM and composite kernels are defined and explained in details in (Melgani and Bruzzone 2004; Camps-Valls et al. 2006). For comparison, the results obtained from SVM with composite kernels are also shown in Fig. 3(b) and Fig. 4(b). The ROC curves for the two test HYDICE images using SMF, MSD, ASD, and SVM are shown in Fig. 2. Overall from this figure, one can observe that for both images the detector based on the joint sparsity model yields the best performance. The simultaneous sparse recovery algorithms clearly outperforms the classical target detection/classification algorithms.

Next we show the effect of neighborhood size on the detection performance. In this experiment, the simultaneous sparse approximation problem in (10) is solved by SOMP for different neighborhood at a fixed sparsity level $K_0 = 10$. Specifically, we use 1×1 neighborhood (equivalent to the pixel-wise sparsity model), 4-connected neighborhood, 3×3, 5×5, and 7×7 window neighborhood, corresponding to $T = 1$, 5, 9, 25, and 49, respectively. The ROC curves for the

various types of neighborhood are shown in Fig. 5. By incorporating the contextual interaction between neighboring pixels, the detector performance is significantly improved. As the neighborhood size increases, the performance tends to saturate.

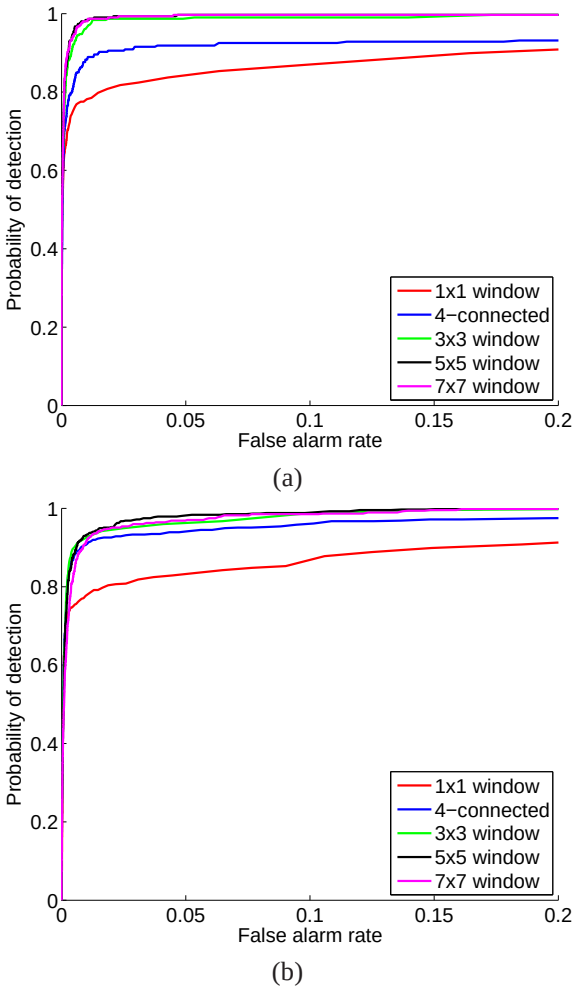


Fig. 5. Effects of the neighborhood size T on detection performance for (a) DR-II and (b) FR-I.

Next we demonstrate how the detection results are affected by the sparsity level of the representation. The sparsity level refers to the number of nonzero rows in $\boldsymbol{S}$, which is also the number of common atoms selected from the dictionary by the greedy algorithms to simultaneously approximate all of the neighboring pixels. The neighborhood size T is fixed to 25 (i.e., a 5×5 window is used) in this experiment. The ROC curves for both images using SOMP with sparsity levels $K = 1$, 2, 3, 5, 10, and 15 are shown in Fig. 6. For very small K, the sparsity-based technique recovery problem is reduced to a simple template matching and leads to underfitting. Generally the detection performance improves as the sparsity level K increases to a certain level. However, if K is too large, the solution becomes dense and involves both background and target atoms, and thus it loses its discriminative power. In this letter, the sparsity level K is chosen to be slightly smaller than the size of the target dictionary.

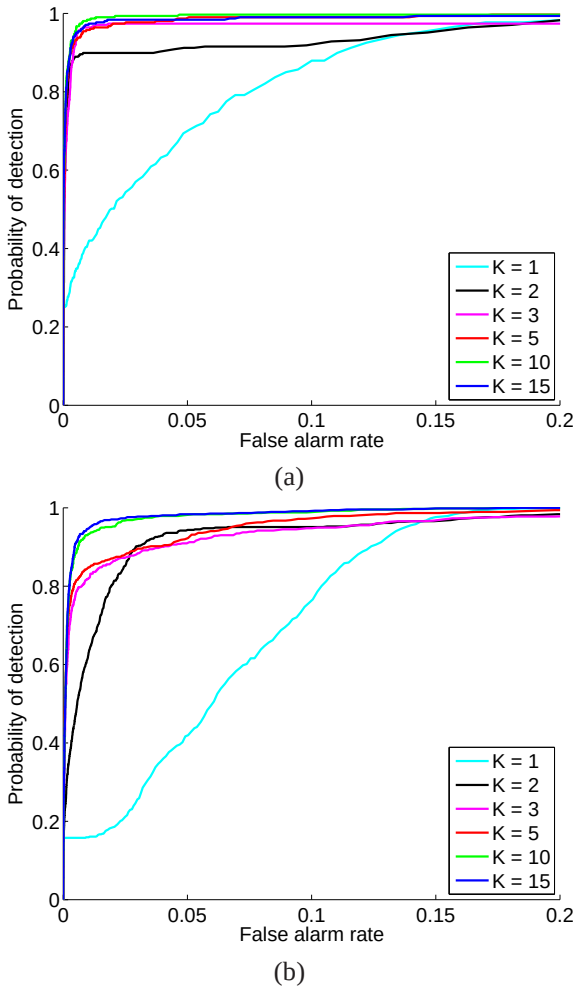


Fig. 6. Effects of the sparsity level K on detection performance for (a) DR-II and (b) FR-I.

4. CONCLUSIONS

In this letter, a joint sparsity model is proposed for target detection in HSI. The inter-pixel correlation in HSI is incorporated by employing the joint-sparsity model where pixels in a small neighborhood in the test image are represented by a linear combination of a few common training samples weighted with a different set of coefficients for each pixel. The sparse representations of the neighboring pixels are obtained by a simultaneous sparse recovery algorithm SOMP. Then resulting sparse representations are then used directly for target detection. Simulation results show that the proposed algorithm outperforms the classical hyperspectral target detection algorithms, including SMF, MSD, ASD, and SVM.

REFERENCES

Basedow, R. W., D. C. Carmer, and M. E. Anderson, 1995: HYDICE system: Implementation and performance. *Proc. SPIE Conference on Algorithms and Technologies for Multispectral, Hyperspectral, and Ultraspectral Imagery XV.*, Vol. 2480, 258–267.

Bruckstein, A. M., D. L. Donoho, and M. Elad, 2009: From sparse solutions of systems of equations to sparse modeling of signals and images. *SIAM Review*, **51 (1)**, 34–81.

Camps-Valls, G., L. Gomez-Chova, J. Muñoz-Marí, J. Vila-Francés, and J. Calpe-Maravilla, 2006: Composite kernels for hyperspectral image classification. *IEEE Geoscience and Remote Sensing Letters*, **3 (1)**, 93–97.

Chen, Y., N. M. Nasrabadi, and T. D. Tran, 2010: Sparse subspace target detection for hyperspectral imagery. *Proc. SPIE Conference on Algorithms and Technologies for Multispectral, Hyperspectral, and Ultraspectral Imagery XVI*, Vol. 7695.

Cotter, S. F., B. D. Rao, K. Engan, and K. Kreutz-Delgado, 2005: Sparse solutions to linear inverse problems with multiple measurement vectors. *IEEE Trans. on Signal Processing*, **53 (7)**, 2477–2488.

Kwon, H. and N. M. Nasrabadi, 2007: A comparative analysis of kernel subspace target detectors for hyperspectral imagery. *EURASIP Journal on Applied Signal Processing*, **2007 (1)**, 193–193.

Manolakis, D. and G. Shaw, 2002: Detection algorithms for hyperspectral imaging applications. *IEEE Signal Processing Magazine*, **19 (1)**, 29–43.

Melgani, F. and L. Bruzzone, 2004: Classification of hyperspectral remote sensing images with support vector machines. *IEEE Trans. on Geoscience and Remote Sensing*, **42 (8)**, 1778–1790.

Rakotomamonjy, A., submitted for publication, 2010: Surveying and comparing simultaneous sparse approximation (or group-lasso) algorithms. *Elsevier*.

Rudin, L., S. Osher, and E. Fatemi, 1992: Nonlinear total variation based noise removal algorithms. *Physica D.*, **60**, 259–268.

Tropp, J. and A. Gilbert, 2007: Signal recovery from random measurements via orthogonal matching pursuit. *IEEE Trans. on Information Theory*, **53 (12)**, 4655–4666.

Tropp, J. A., A. C. Gilbert, and M. J. Strauss, 2006: Algorithms for simultaneous sparse approximation. Part I: Greedy pursuit. *Signal Processing, special issue on Sparse approximations in signal and image processing*, **86**, 572–588.

Tropp, J. A. and S. J. Wright, 2010: Computational methods for sparse solution of linear inverse problems. *Proceedings of the IEEE*, **98 (6)**, 948–958.

Wright, J., Y. Ma, J. Mairal, G. Sapiro, T. Huang, and S. Yan, 2010: Sparse representation for computer vision and pattern recognition. *Proceedings of the IEEE*, **98 (6)**, 1031–1044.