Regularized Multiple Sparse Bayesian Learning for Hyperspectral Target Detection

Fanqiang Kong[1] · Keyao Wen[1] · Yunsong Li[2]

Abstract

The sparse representation method has been successfully applied in the field of hyperspectral image target detection. It assumes that target detection can be achieved by using target and background libraries to represent test pixels. Under this formulation, the presentation of the target and background signatures can be solved by L_1-norm minimization of the weight coefficient and the target detection output is simply achieved by the difference between the two representation residuals. In this paper, a regularized multiple sparse Bayesian learning (RMSBL) method for hyperspectral target detection is proposed, which is established by Bayesian inference using the conditional posterior distributions of the model parameters under a hierarchical Bayesian model. According to the cost function for multiple sparse Bayesian learning, the presentation of the target and background signatures can be obtained by an $L_{2,1}$-norm iterative minimization method. And the target detection result can be achieved with the difference between the two representation residuals. Four groups of hyperspectral datasets are used for simulation experiments. The results are compared with those of other common detection algorithms. The experimental results demonstrate that the RMSBL algorithm has higher detection performance.

Keywords Hyperspectral imagery · Target detection · Sparse representation · Multiple sparse Bayesian learning

Introduction

Target detection, as an important research direction in the field of hyperspectral imaging, aims to detect small objects or anomalies in hyperspectral image (HSI). It has broad application potential in military safety, environmental pollution monitoring, geological exploration, agriculture and forest monitoring, etc. (Matteoli et al. 2010; Li et al. 2017; Xie et al., 2019a, b). HSI contains rich spectral information and spatial information. Since spectral features differ among substances, it is possible to effectively distinguish objects in a scene using HSI.

✉ Fanqiang Kong
kongfq@nuaa.edu.cn

Keyao Wen
wkynuaa@163.com

Yunsong Li
ysli@mail.xidian.edu.cn

[1] College of Astronautics, Nanjing University of Aeronautics and Astronautics, Nanjing 210016, China

[2] State Key Laboratory of Integrated Service Networks, Xidian University, Xi'an 710071, China

Many target detection algorithms have been proposed and applied to a HSI. The Reed-Xiaoli (RX) detector (Reed and Yu, 1990; Xie et al., 2019a, b) obtains the detection result by constructing the generalized likelihood ratio and estimating the background covariance matrix. However, the RX algorithm ignores the rich nonlinear information in a HSI, resulting in poor detection accuracy. The collaboration representation detector (CRD) (Li and Du 2015) is directly based on the concept that each pixel in the background can be approximately represented by its spatial neighborhoods, while anomalies cannot but it only considers the spectral features of a HSI, ignoring the spatial features.

The combined sparse and collaboration representation detector (CSCR) (Li et al. 2015) assumes that the representation of known target signatures is sparse and can be solved by L_1-norm minimization of the representation weight vector. However, the representation of background atoms is assumed to be collaborative and can be solved by L_2-norm minimization. Finally, the decision can be made by computing the difference between the two representation residuals.

Support vector machines (SVM) (Zhao et al. 2012) are a highly effective method for nonlinear signals. It maps the signals to a new feature space where it is easier to distinguish among signals (Tan and Du 2008). The kernel method has

yielded satisfactory results in HSI processing (Li et al. 2010; Zhao et al. 2010). In addition, many algorithms use statistics for hypothesis detection, such as spectral matched filter (SMF) (MANOLAKIS and SHAW 2002), all of which must assume the mathematical distribution of the pixel spectrum of hyperspectral images (HSIs). The accuracy of the hypothesis distribution model has a substantial impact on the detection results.

In recent years, the sparse representation method has attracted increasing attention (Chen et al., 2011a, b). It assumes that target detection can be achieved by using target and background libraries to represent test pixels. Under this formulation, the presentation of the target and background signatures can be solved by L_1-norm minimization of the weight coefficient. The target detection problem is thus transformed into the optimization problem of solving for dictionary atomic coefficients (Chen et al., 2011a, b). This algorithm does not require assumptions regarding the mathematical distribution model of the background and target pixels, nor does it require the atoms of the training sample dictionary to be independent of one another. Because of the characteristics of background and target pixel spectra, they belong to different subspaces and the atoms that constitute their spectral dictionaries will differ. Whether the pixel belongs to the background or the target can be determined according to the positions of dictionary atoms with nonzero sparseness.

For solving the dictionary atomic optimization problem, convex optimization methods typically use the L_1-norm instead of the L_0-norm to solve the problem efficiently. However, only under strict conditions do the two yield equivalent solutions and the actual signal typically cannot be recovered. In addition, due to the similarity of the endmembers in the spectral library and the insufficient constraints of the optimization function, the real solutions differ substantially from the results of the abundance estimation. At the same time, because of the large amount of hyperspectral image data, the convex optimization algorithm is slow.

Another method is sparse Bayesian learning (SBL) (Themelis et al. 2012) that models the unknown variables based on Bayesian concepts and obtains sparse solutions by Bayesian derivation (Wipf 2006; Zhang and Rao 2012, 2013; Qiu and Dogandzic 2010; Zhang and Rao 2011; Kong et al. 2017). The core strategy of Bayesian theory is to obtain the probability of an a posteriori unknown parameter by using the sample prior information and the complete information. Wipf and Rao (2004) proved that the SBL algorithm can obtain the sparsest solution. SBL still performs reliably if the endmembers in the spectral library are strongly correlated. However, the expectation maximization (EM) algorithm is used to update the parameters. This leads to a large amount of computation and fails to consider the joint sparsity of the endmember combinations in adjacent pixels, thereby resulting in low efficiency of the algorithm.

To make better use of the spatial correlation of HSI, a regularized multiple sparse Bayesian learning (RMSBL) method for target detection in HSI is proposed, that is established by Bayesian inference using the conditional posterior distribution of the model parameters under a hierarchical Bayesian model. According to the cost function for multiple sparse Bayesian learning (MSBL), the presentation of the target and background signatures can be obtained by the $L_{2,1}$-norm minimization iterative method and the target detection result can be achieved with the difference between the two representation residuals. In the simulation experiment, compared with other commonly used detection algorithms, the RMSBL algorithm has superior detection performance.

Target Detection Based on Sparse Representation

Let **Y** be a set of hyperspectral image data and $\mathbf{y} \in R^{N \times 1}$ be an N-dimensional spectral vector in **Y**. Vector **y** can be represented as follows (Zhang et al. 2017):

$$
\begin{aligned}
\mathbf{y} &= \mathbf{a}_1^b x_1^b + \mathbf{a}_2^b x_2^b + \cdots + \mathbf{a}_{N_b}^b x_{N_b}^b + \mathbf{a}_1^t x_1^t + \mathbf{a}_2^t x_2^t + \cdots + \mathbf{a}_{N_t}^t x_{N_t}^t + \mathbf{n} \\
&= \left[\mathbf{a}_1^b \mathbf{a}_2^b \cdots \mathbf{a}_{N_b}^b \right] \left[x_1^b x_2^b \cdots x_{N_b}^b \right]^T + \left[\mathbf{a}_1^t \mathbf{a}_2^t \cdots \mathbf{a}_{N_t}^t \right] \left[x_1^t x_2^t \cdots x_{N_t}^t \right]^T + \mathbf{n} \\
&= \mathbf{A}_b \mathbf{x}_b + \mathbf{A}_t \mathbf{x}_t + \mathbf{n} = [\mathbf{A}_b \mathbf{A}_t] \begin{bmatrix} \mathbf{x}_b \\ \mathbf{x}_t \end{bmatrix} + \mathbf{n} = \mathbf{A}\mathbf{x} + \mathbf{n}
\end{aligned}
\tag{1}
$$

where $\mathbf{A}_b$ and $\mathbf{A}_t$ are the background dictionary and the target dictionary, respectively; **x** denotes the weight coefficients that correspond to the dictionary; **x** is a sparse vector, of which only a few coefficients are nonzero; and **n** represents the observation error.

In the sparse model, it is not necessary to assume the distribution characteristics of the target and the background because the spectral characteristics of the background and target pixels differ and are distributed in different subspaces. The sparse vector x is composed of background weight coefficient $\mathbf{x}_b$ and target weight coefficient $\mathbf{x}_t$. If **y** is a target pixel, then $\mathbf{x}_b$ is a zero vector and $\mathbf{x}_t$ is a sparse vector; if **y** is a background pixel, then $\mathbf{x}_b$ is a sparse vector and $\mathbf{x}_t$ is a zero vector. Therefore, based on the nonzero position of the coefficient **x** of pixel **y**, whether the pixel is a background or a target pixel can be determined.

To obtain the weight coefficient **x** of the pixel **y**, one must solve the optimization problem that is defined by the following formula:

$$
\mathbf{x} = \mathrm{argmin} \|\mathbf{x}\|_1 \text{ subject to } \mathbf{A}\mathbf{x} = \mathbf{y}
\tag{2}
$$

where $\mathrm{argmin}(x)$ represents the value of the variable when the objective function takes the minimum value.

Pixel **y** can be classified by comparing the values of the reconstructed residuals after **x** has been obtained. Therefore, the output of the detector is expressed as follows:

$$R(\mathbf{y}) = \|\mathbf{y}-\mathbf{A}_b\mathbf{x}_b\|_2 - \|\mathbf{y}-\mathbf{A}_t\mathbf{x}_t\|_2 \tag{3}$$

where $\|\mathbf{y}-\mathbf{A}_b\mathbf{x}_b\|_2$ and $\|\mathbf{y}-\mathbf{A}_t\mathbf{x}_t\|_2$ are the background residual and the target residual, respectively. For a specified threshold δ, if $R(\mathbf{y}) > \delta$, then **y** is classified as a target; otherwise, **y** belongs to the background.

The above is a target detection model that is based on sparse representation. Because the adjacent pixels in hyperspectral data contain similar information, they can be linearly combined by mixing coefficients of the endmembers in the spectral library. The spectral vector is extended to a spectral matrix and the weight coefficient vector is expanded to a weight coefficient matrix. Then, the mathematical model of the multiple sparse representation is as follows:

$$\mathbf{Y} = \mathbf{A}\mathbf{X} + \mathbf{N} = \mathbf{A}_t\mathbf{X}_t + \mathbf{A}_b\mathbf{X}_b + \mathbf{N} \tag{4}$$

where $\mathbf{Y} \in R^{N \times M}$ denotes the observed values of M pixels in N bands; $\mathbf{A} \in R^{N \times L}$ represents the spectral library, which contains the reflected values of L objects in N bands; $\mathbf{X} \in R^{L \times M}$ is the weight coefficient matrix; and **N** is the observation noise.

The key step in target detection is to find weight coefficient **X** such that $\|\mathbf{Y}-\mathbf{A}\mathbf{X}\|_F^2$ is minimized under the constraint that $\|\mathbf{X}\|_{2,1}$ is also minimized. Therefore, the objective function is as follows:

$$\mathbf{X} = \arg\min_{\mathbf{X}} \|\mathbf{Y}-\mathbf{A}\mathbf{X}\|_F^2 + \|\mathbf{X}\|_{2,1} \tag{5}$$

The optimization formula of the $L_{2,1}$-norm is typically solved by a convex optimization algorithm. The representative algorithm for convex optimization is the collaborative spectral unmixing by variable splitting and augmented Lagrangian algorithm (CLSUnSAL) (Bioucas-Dias and Figueiredo 2010).

The values of the target and the background under the sparse representation can be obtained separately after solving for the weight coefficients:

$$\mathbf{Y}_t = \mathbf{A}_t\mathbf{X}_t \tag{6}$$

$$\mathbf{Y}_b = \mathbf{A}\mathbf{X}-\mathbf{Y}_t \tag{7}$$

Then, the residuals of the pixels are calculated one by one, and the target residual and background residual are calculated as follows:

$$r_t(\mathbf{y}) = \|\mathbf{y}-\mathbf{y}_t\|_2^2 \tag{8}$$

$$r_b(y) = \|y-y_b\|_2^2 \tag{9}$$

Pixel **y** is classified by comparing the values of the reconstructed residuals. Therefore, the output of the detector is expressed as follows:

$$R(\mathbf{y}) = r_b(\mathbf{y})-r_t(\mathbf{y}) \tag{10}$$

For a specified threshold δ, if $R(\mathbf{y}) > \delta$, then **y** is classified as a target; otherwise, **y** belongs to the background.

Regularized Multiple Sparse Bayesian Learning for Target Detection

Multiple Sparse Bayesian Learning Model

Multiple sparse Bayesian learning is an efficient method for solving the simultaneous sparse approximation problem in the simultaneous sparse model. Based on the MMV model, the prior distribution of the sparse coefficient matrix with joint hyperparameters is established. Because the current pixels in hyperspectral images and the surrounding pixels contain similar information, the coefficient matrix should satisfy the row sparsity characteristic. The prior distribution of each row vector is characterized by hyperparameters such that the matrix satisfies the row sparsity characteristic. According to the cost function of MSBL, an iterative method is obtained by theoretical derivation, which effectively reduces the number of iterations.

Let $\mathbf{Y}_{\cdot j}$ and $\mathbf{X}_{\cdot j}$ represent the jth columns of **Y** and **X**, respectively. The likelihood function is obtained (Kong et al. 2016):

$$p\left(\mathbf{Y}_{\cdot j}|\mathbf{X}_{\cdot j}\right) = \left(\pi\sigma^2\right)^{-N}\exp\left(-\frac{1}{\sigma}\left\|\mathbf{Y}_{\cdot j}-\mathbf{A}X_{\cdot j}\right\|_2^2\right) \tag{11}$$

The general method is to represent the sparsity of the abundance matrix directly by Laplace a priori, which will cause the likelihood function and prior distribution not to satisfy the requirement of conjugation. Therefore, the hierarchical Bayesian model is used to design the prior distribution. Assume that the ith row $\mathbf{X}_{i\cdot}$ of the coefficient matrix **X** obeys the Gaussian distribution $p(\mathbf{X}_{i\cdot}; \gamma_i) = N(0, \gamma_i I)$ of the parameter γ_i. Then, the prior distribution of the coefficient matrix is a high-dimensional Gaussian distribution:

$$p(\mathbf{X}; \gamma) = \prod_{i=1}^{M} p(\mathbf{X}_{i\cdot}; \gamma_i) \tag{12}$$

where $\gamma = [\gamma_1, \gamma_2, \cdots, \gamma_M]^T \in R_+^M$, γ_i is used to denote the sparsity of each row of the coefficient matrix. If $\gamma_i = 0$, $\mathbf{X}_{i\cdot}$ is all zero rows, namely, the conditional probability $p(\mathbf{X}_{i\cdot} = 0| \mathbf{Y}; \gamma_i = 0) = 1$ is satisfied. The parameter γ_i obeys the Gamma distribution: $p(\gamma_i| \lambda_i) \sim \Gamma(\gamma_i| 1, \lambda_i/2)$.

According to Bayesian theory, a posterior distribution is obtained:

$$p(\mathbf{X}_{.j}|\mathbf{Y}_{.j};\gamma) = \frac{p(\mathbf{X}_{.j},\mathbf{Y}_{.j};\gamma)}{\int p(\mathbf{X}_{.j},\mathbf{Y}_{.j};\gamma)d\mathbf{X}_{.j}} = N(\mu_{.j},\Sigma) \tag{13}$$

The mean and variance can be expressed as follows:

$$\mathbf{M} = [\mu_{.1},\mu_{.2},\cdots,\mu_{.L}] = E[\mathbf{X}|\mathbf{Y};\gamma] = \Gamma\mathbf{A}^T\textstyle\sum_{\mathbf{Y}}^{-1}\mathbf{Y} \tag{14}$$

$$\textstyle\sum = Cov[\mathbf{X}_{.j},\mathbf{Y}_{.j};\gamma] = \Gamma - \Gamma\mathbf{A}^T\textstyle\sum_{\mathbf{Y}}^{-1}\mathbf{A}\Gamma, \forall j \tag{15}$$

where $\Gamma = diag(\gamma)$ and $\sum_{\mathbf{Y}} = \sigma^2 I + \mathbf{A}\Gamma\mathbf{A}^T$.

The logarithmic form of the cost function of MSBL is expressed as follows (Wang et al. 2013):

$$L(\gamma,\sigma^2) = -2\log\int p(\mathbf{Y}|\mathbf{X})p(\mathbf{X};\gamma,\sigma^2)d\mathbf{X} = L\log|\textstyle\sum_{\mathbf{Y}}| + \textstyle\sum_{j=1}^{L}\mathbf{Y}_{.j}^T\textstyle\sum_{\mathbf{Y}}^{-1}\mathbf{Y}_{.j} \tag{16}$$

For parameter optimization, the EM method is commonly used in sparse Bayesian learning. The EM method is divided into two steps: In step E, the mean value is calculated via formula (14) and in step M, the iterative update is carried out by the following formula:

$$\gamma_i = \mu_i^2 + \textstyle\sum_{ii} \tag{17}$$

The MacKay method obtains the parameter iteration formula by calculating the extremum. This method is equivalent to point estimation and has a faster iteration speed than formula (17):

$$\gamma_i = \frac{\mu_i^2}{1 - \gamma_i^{-1}\sum_{ii}} \tag{18}$$

Optimal Solution

This paper proposes a new method for optimization. In Eq. (16), the former term, namely, $\log|\sum_{\mathbf{Y}}|$, is a smooth concave function, which can be transformed by the property of conjugate functions, while the latter term, namely, $\mathbf{Y}_{.j}^T\sum_{\mathbf{Y}}^{-1}\mathbf{Y}_{.j}$, is a quadratic term. After deducing the two terms, the following is obtained:

$$L_z(\mathbf{X},\gamma) = \frac{1}{\sigma^2}\min_{\mathbf{X}}\|\mathbf{Y} - \mathbf{AX}\|_F^2 + z^T\gamma + \mathbf{X}^T\Gamma^{-1}\mathbf{X}$$

$$= \frac{1}{\sigma^2}\min_{\mathbf{X}}\|\mathbf{Y} - \mathbf{AX}\|_F^2 + z^T\gamma + \textstyle\sum_{i=1}^{M}\gamma_i^{-1}\|\mathbf{X}_{i.}\|_2^2 \tag{19}$$

where min(x) represents a function that takes the minimum value of the objective function.

The optimal iteration of γ_i is obtained by using the derivation rule for Eq. (19):

$$\gamma_i = z_i^{-1/2}\sqrt{\mathbf{X}_{i.}\mathbf{X}_{i.}^T} = z_i^{-1/2}\|\mathbf{X}_{i.}\|_2 (\forall i) \tag{20}$$

Formula (20) is substituted for formula (19) and the coefficients of the regular terms are normalized:

$$\mathbf{X} = \underset{\mathbf{X}}{\operatorname{argmin}}\frac{1}{2}\|\mathbf{Y} - \mathbf{AX}\|_F^2 + \textstyle\sum_{i=1}^{M}\sigma^2\gamma_i^{-1}\|\mathbf{X}_{i.}\|_2^2 \tag{21}$$

Let $w_i = \sigma^2 z_i^{1/2}$. The optimal expression for weight coefficient estimation is as follows:

$$\mathbf{X} = \underset{\mathbf{X}}{\operatorname{argmin}}\frac{1}{2}\|\mathbf{Y} - \mathbf{AX}\|_F^2 + \textstyle\sum_{i=1}^{M}\sigma^2 z_i^{1/2}\|\mathbf{X}_{i.}\|_2$$

$$= \arg\min_{\mathbf{X}}\frac{1}{2}\|\mathbf{Y} - \mathbf{AX}\|_F^2 + \textstyle\sum_i^M w_i\|\mathbf{X}_{i.}\|_2 \tag{22}$$

Define the matrix norm:

$$\|B\|_{2,1} = \textstyle\sum_i\sqrt{\textstyle\sum_j(B_{ij}^2)} = \textstyle\sum_i\sqrt{B_{i.}B_{i.}^T} = \textstyle\sum_i\|B\|_2 \tag{23}$$

Equation (22) can be rewritten as follows:

$$\mathbf{X} = \arg\min_{\mathbf{X},\,W}\frac{1}{2}\|\mathbf{Y} - \mathbf{AX}\|_F^2 + \|WX\|_{2,1} \tag{24}$$

where $W = \operatorname{diag}(w_i)$ denotes a diagonal matrix with diagonal elements w_i. Equation (24) is an $L_{2,1}$-regularization weighted iterative problem. The results that are obtained by the alternating iteration method are globally convergent and are the sparsest solution (Rakotomamonjy 2011).

In addition, since the noise variance only affects the convergence speed, it does not affect the accuracy of the sparse solution. To set the parameter value adaptively, we update the calculation of the variance as follows (Wipf and Rao 2007):

$$(\sigma^2) = \frac{\|\mathbf{Y} - \mathbf{AX}\|_F^2/L}{N - M + \textstyle\sum_{i=1}^{M}\sum_{ii}/\gamma_i} \tag{25}$$

For problem (16), parameter learning can be performed by alternate iteration; the expressions for which are listed in Table 1.

The overall RMSBL algorithm is summarized as Algorithm 1.

Experimental Results and Analysis

In this section, experiments are conducted on four datasets, and we compare the results with five widely used methods: the CRD, CSCR, RX, LRX, and RMSBL algorithms. The parameters of each algorithm are optimized in the experiment, and the receiver operating characteristic (ROC) curve is typically

Table 1 Parameter updating in each iteration

1. $\forall i \,, \gamma_i^{(k+1)} \leftarrow \left(z_i^{(k)}\right)^{-1/2} \left\|\mathbf{X}_{i\cdot}^{(k)}\right\|_2$

2. $\forall i, z_i^{(k+1)} \leftarrow \mathbf{A}_i^T \left(\sigma^2 I + \mathbf{A}\varGamma^{(k+1)}\mathbf{A}^T\right)^{-1}\mathbf{A}_i$

3. $\left(\sigma^2\right)^{(k+1)} \leftarrow \dfrac{\left\|\mathbf{Y}-\mathbf{A}\mathbf{X}^{(k)}\right\|_F^2/L}{N-M+\sum_{i=1}^{M}\sum_{ii}^{(k+1)}/\gamma_i^{(k+1)}}$

4. $\forall i, w_i^{(k+1)} \leftarrow \left(\sigma^2\right)^{(k+1)}\left(z_i^{(k+1)}\right)^{1/2}$

employed to quantitatively evaluate the detection performance. Then, we compute the area under the ROC curve (AUC) to evaluate the performance of the RMSBL and other algorithms.

Hyperspectral Data

The first experimental dataset, namely, airport2, uses a portion of the hyperspectral image of Los Angeles Airport that was collected by the airborne visible/infrared imaging spectrometer (AVIRIS) sensor. This scene consists of 100×100 pixels (as shown in Fig. 1a) and the spatial resolution is 7.1 m. After removing the water absorption and low-SNR bands, 205 bands remain, including 87 target pixels to be detected. Figure 1 b shows the ground-truth image of the target.

The second dataset, namely, Cuprite, was captured by the AVIRIS sensor in 1997 over the Cuprite mine in Nevada. Only a small part of the data is used in this experiment. Figure 1 c and d show the color composites of Cuprite and the ground-truth image of the target, respectively. This scene consists of 250×191 pixels. After removing the water absorption and low-SNR bands, 188 bands remain, including approximately 35 to 40 target pixels to be detected.

The third dataset, San Diego, uses a portion of the hyperspectral image of the San Diego Airport in the USA that was collected by the AVIRIS sensor. This scene consists of 200×200 pixels (as shown in Fig. 1e) and the spatial resolution is 3.5 m. After removing the water absorption and low-SNR bands, 189 bands remain, including approximately 132 target pixels to be detected. Figure 1 f shows the ground-truth image of the target.

The fourth dataset, the HYDICE Urban scene, is a hyperspectral image of a suburban residential area in Texas, USA, that was captured by Hyperspectral Digital Imagery Collection Experiment (HYDICE) sensor. Figure 1 g and h show the color composites of the HYDICE Urban scene and the ground-truth image of the target, respectively. This scene consists of 80×100 pixels, and the spatial resolution is approximately 1 m. After removing the water absorption and low-SNR bands, 162 bands remain, including approximately 21 target pixels to be detected.
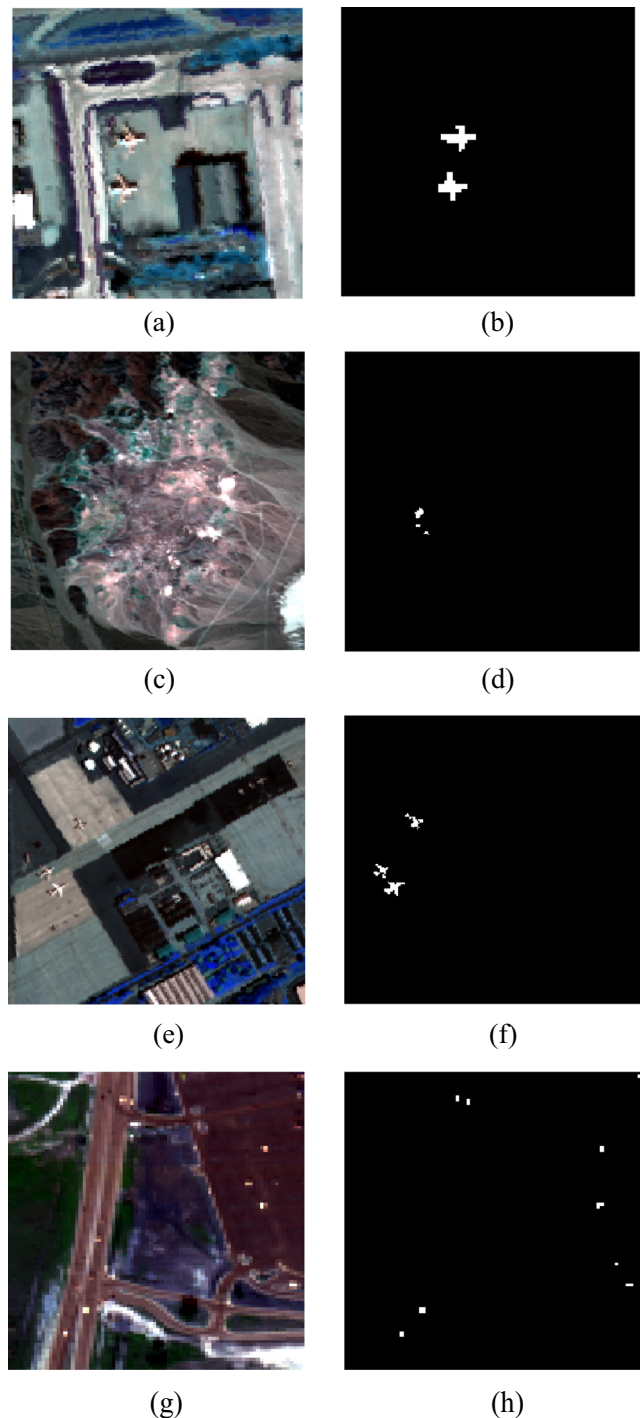

(a)   (b)

(c)   (d)

(e)   (f)

(g)   (h)

Fig. 1 The first column shows the color composites of four datasets and the second column shows the ground-truth map of the target. **a**, **b** Airport2. **c**, **d** Cuprite. **e**, **f** San Diego. **g**, **h** The HYDICE Urban scene

Detection Performance

According to the existing theoretical knowledge, algorithms RMSBL, CRD, CSCR, and LRX all use a background dictionary. In the actual target detection, the background dictionary is typically obtained via local and

Fig. 2 Detection outputs for dataset airport2. **a** RMSBL. **b** CRD. **c** CSCR. **d** LRX. **e** RX

(a)　　　　　(b)　　　　　(c)

(d)　　　　　(e)
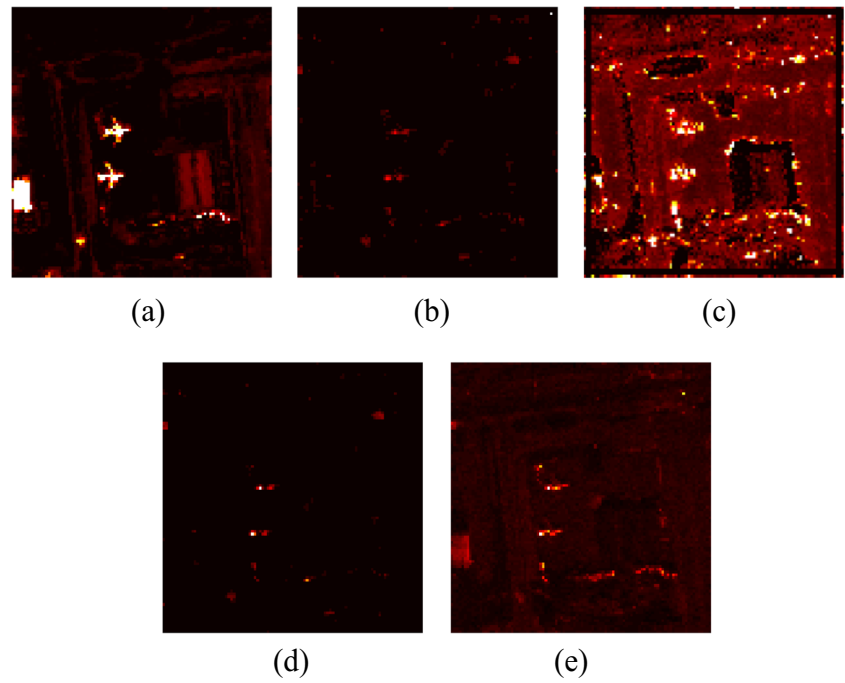
adaptive methods. For algorithms CRD, CSCR, and LRX, the sliding dual-window method is used to obtain the background dictionary; hence, the sizes of the inner and outer windows will affect the performance of the algorithm. Therefore, the performance of the algorithm is optimized by adjusting the parameters during the simulation experiment. The RMSBL algorithm obtains the dictionary via the vertex component analysis (VCA) method; thus, the RMSBL and RX algorithms are not affected by the dual-window scheme.

Fig. 3 Detection outputs for the Cuprite dataset. **a** RMSBL. **b** CRD. **c** CSCR. **d** LRX. **e** RX
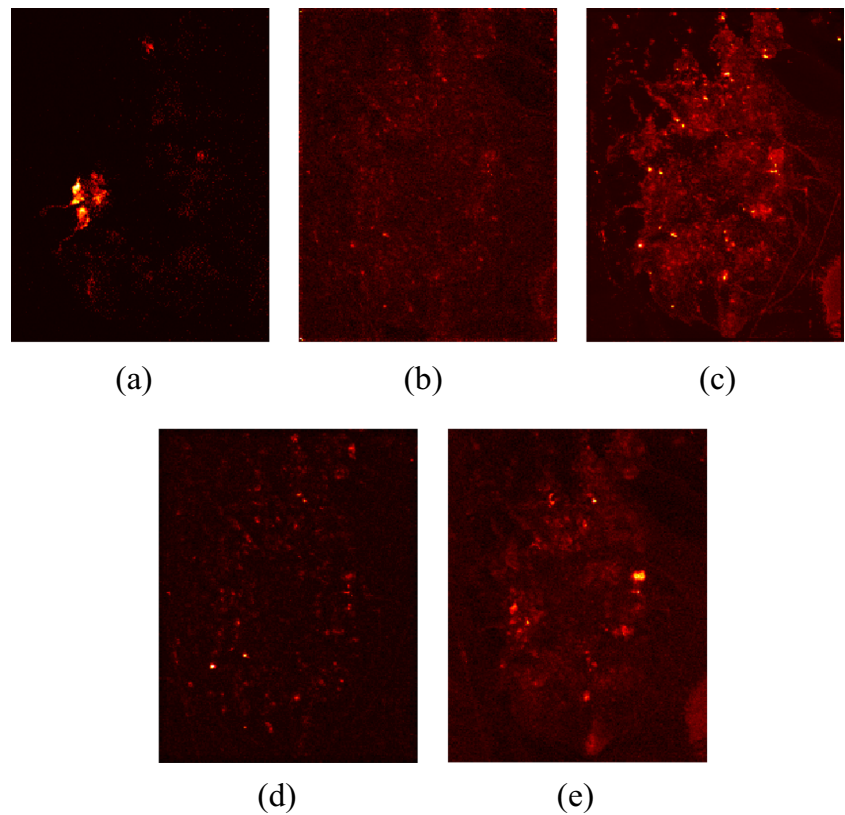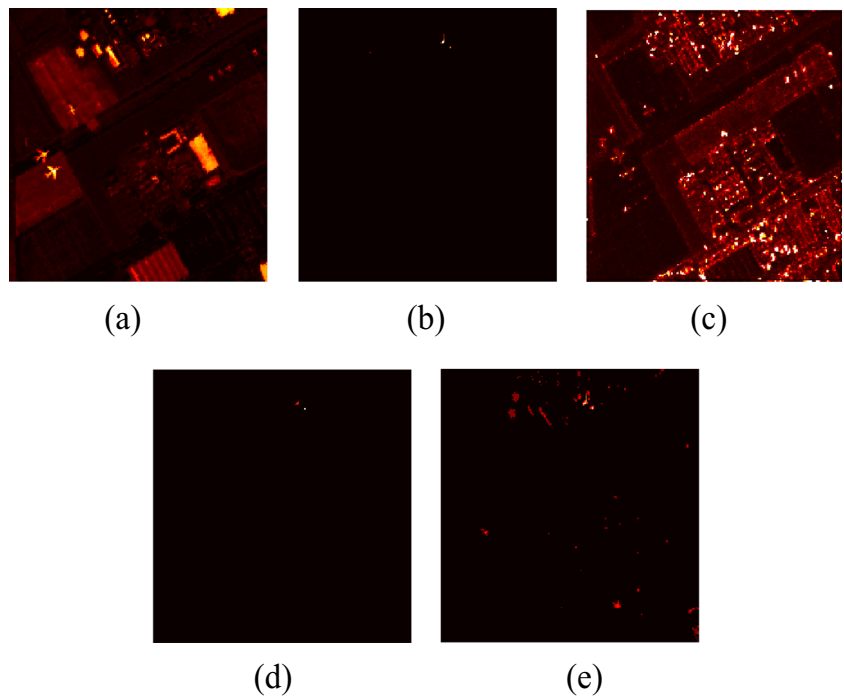

(a)　　　　　(b)　　　　　(c)

(d)　　　　　(e)

Fig. 4 Detection outputs for the San Diego dataset. **a** RMSBL. **b** CRD. **c** CSCR. **d** LRX.. **e** RX


(a)   (b)   (c)

(d)   (e)

We conduct an experimental simulation on the four datasets that are described above and analyze the results of the RMSBL algorithm and the four comparison algorithms in terms of their ROC curves and AUC values.

To optimize the performance of the algorithm for the airport2 dataset, the parameters of each algorithm are determined after many experiments as follows: for the CRD algorithm, the outer window size is $w_{out} = 11$, the inner window size is $w_{in} = 5$, and the regularization parameter is $\lambda = 10^{-6}$; for the CSCR algorithm, the window sizes are $(w_{out}, w_{in}) = (11, 3)$, and the regularization parameter is $\lambda_1 = 10^{-2}$, $\lambda_2 = 10^{-1}$; and for the LRX algorithm, the window sizes are $(w_{out}, w_{in}) = (15, 3)$. The

Fig. 5 Detection outputs for the HYDICE Urban scene dataset. **a** RMSBL. **b** CRD. **c** CSCR. **d** LRX. **e** RX
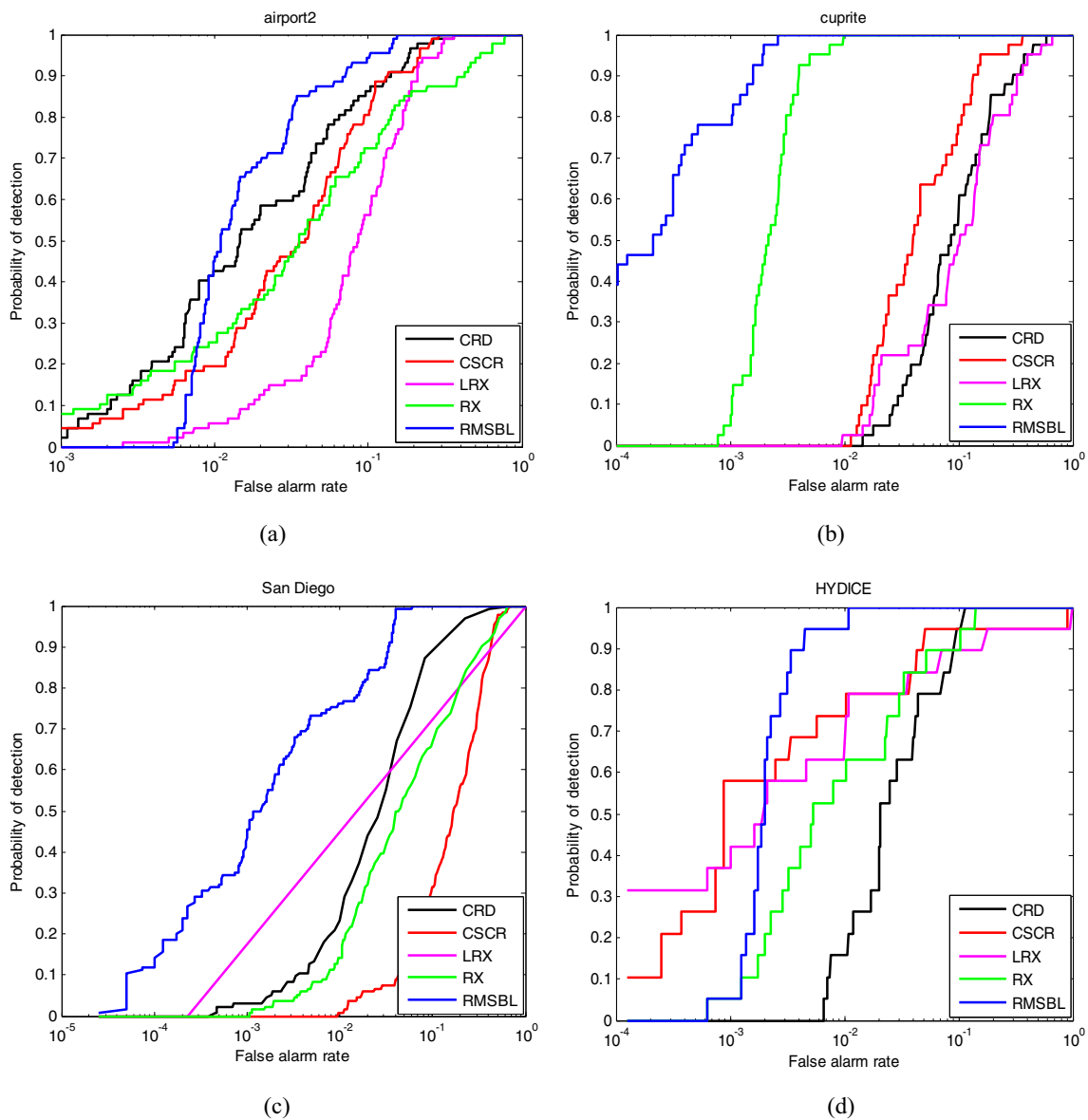

(a)   (b)   (c)

(d)   (e)
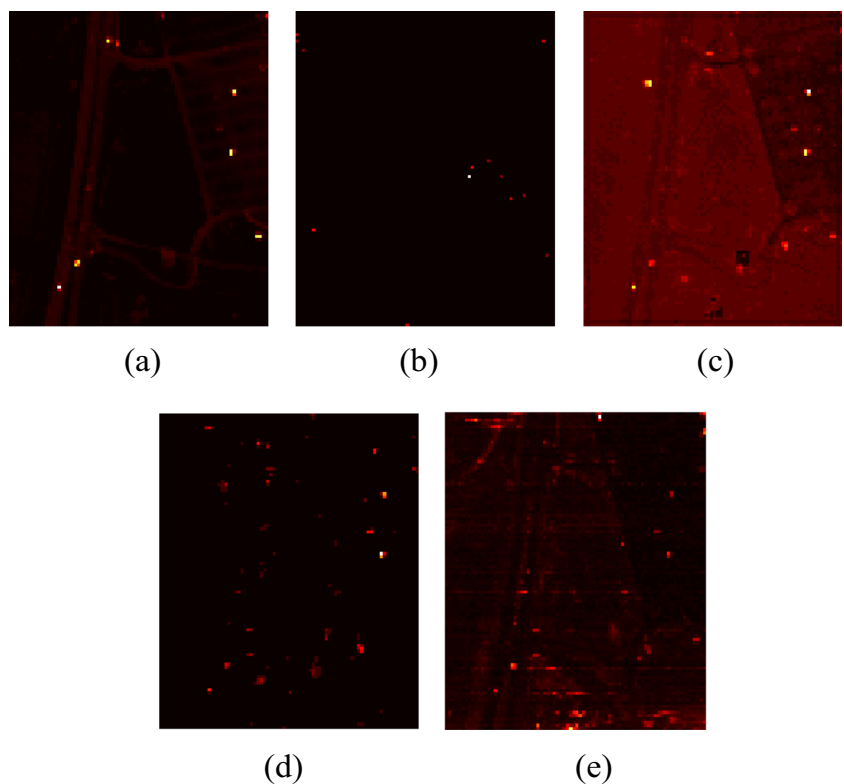
Fig. 6 ROC performance of the proposed method. **a** Airport2 dataset. **b** Cuprite dataset. **c** San Diego dataset. **d** HYDICE dataset

detection outputs of the algorithms are shown in Fig. 2. The proposed RMSBL algorithm yields the best result. Figure 6 a shows the ROC curves of the proposed algorithm and the comparison algorithms. The detection rate of the RMSBL algorithm is lower than those of the other algorithms if the

false-alarm rate is less than 10^{-2}; however, it increases rapidly if the false alarm rate exceeds 10^{-2}, significantly higher compared with the other algorithms; and it reaches 1 before those of the other algorithms. On this dataset, the CRD algorithm is inferior to the RMSBL but outperforms the other algorithms.

Table 2   AUC (%) values for the proposed algorithm and the comparison algorithms

	Airport2	Cuprite	San Diego	HYDICE
CRD	95.55	87.08	95.12	96.41
CSCR	94.25	93.27	79.42	94.42
LRX	89.49	85.33	49.99	93.24
RX	88.95	99.74	88.64	97.63
RMSBL	97.47	99.95	99.18	99.75

Table 3   Execution times (in seconds) on all experimental datasets

	Airport2	Cuprite	San Diego	HYDICE
CRD	94.57	398.92	625.90	63.52
CSCR	333.29	1236.83	1107.13	158.03
LRX	97.14	280.91	261.48	40.34
RX	0.18	0.68	0.46	0.09
RMSBL	26.32	195.58	90.80	17.31

For the Cuprite dataset, the parameters of each algorithm are as follows: for the CRD algorithm, the window sizes are $(w_{out}, w_{in}) = (11, 5)$ and the regularization parameter is $\lambda = 10^{-6}$; for the CSCR algorithm, the window sizes are $(w_{out}, w_{in}) = (11, 5)$ and the regularization parameters are $\lambda_1 = 10^{-1}$ and $\lambda_2 = 10^{-2}$; and for the LRX algorithm, the window sizes are $(w_{out}, w_{in}) = (13, 9)$. Figure 3 shows the outputs of the proposed algorithm and the comparison algorithms. The ROC curves of the algorithms are plotted in Fig. 6b. The experimental results demonstrate that the RMSBL algorithm far outperforms the other algorithms in terms of the detection probability and when the false alarm rate is less than 10^{-2}, the detection rate has reached 1; hence, the RMSBL algorithm performs well on this dataset and the RX algorithm is inferior to the RMSBL algorithm but outperforms the other algorithms.

For the San Diego dataset, the parameters of each algorithm are as follows: for the CRD algorithm, the window sizes are $(w_{out}, w_{in}) = (17, 9)$, and the regularization parameter is $\lambda = 10^{-6}$; for the CSCR algorithm, the window sizes are $(w_{out}, w_{in}) = (7, 5)$, and the regularization parameters are $\lambda_1 = 10^{-2}$ and $\lambda_2 = 10^{-1}$; and for the LRX algorithm, the window sizes are $(w_{out}, w_{in}) = (13, 7)$. Figure 4 shows the outputs of the algorithms. Figure 6 c shows the ROC curves of the algorithms, according to which the detection probability of the RMSBL algorithm exceeds those of the other algorithms. If the false alarm rate is close to 10^{-1}, the detection rate reaches 1. Under this set of data, the CRD algorithm is inferior to the RMSBL algorithm but outperforms the other algorithms.

For the HYDICE dataset, the parameters of each algorithm are as follows: for the CRD algorithm, the window sizes are $(w_{out}, w_{in}) = (13, 7)$, and the regularization parameter is $\lambda = 10^{-6}$; for the CSCR algorithm, the window sizes are $(w_{out}, w_{in}) = (9, 5)$, and the regularization parameters are $\lambda_1 = 10^{-2}$ and $\lambda_2 = 10^{-1}$; and for the LRX algorithm, the window sizes are $(w_{out}, w_{in}) = (13, 7)$. Figure 5 shows the outputs of the algorithms. The ROC curves of the algorithms are shown in Fig. 6d. If the false alarm rate is less than 10^{-3}, the detection rate of the RMSBL algorithm is low. If the false alarm rate exceeds 10^{-3}, the detection rate increases rapidly and reaches 1 when the false alarm rate is 10^{-2}.

The AUC values of each algorithm are listed in Table 2. From the data, we can judge the performance of each algorithm more accurately. The AUC value of RMSBL is the largest in the experimental results for each group of data, namely, its performance is the best. For the Cuprite and HYDICE datasets, the AUC value of RX algorithm is slightly smaller than that of RMSBL; however, the RX algorithm does not perform well on the airport2 and San Diego datasets. The RMSBL algorithm performs well on all test datasets, especially the Cuprite and HYDICE datasets, on which the AUC value is close to 1.

Finally, we report the computational complexities of the compared detection methods with optimal parameters. All experiments were conducted using MATLAB R2014a on an Intel Core i5-3470 CPU machine with 12 GB of RAM. The execution times (in seconds) for the experimental data are listed in Table 3. All the other algorithms except the RX algorithm have higher computational costs than RMSBL.

Conclusions

This paper proposes a hyperspectral target detection algorithm that is based on RMSBL. The weight coefficient is calculated by $L_{2,1}$-norm regularization, and the target residual and background residual are obtained. Finally, target detection is achieved by evaluating the difference between the two residuals. The proposed method is compared with the CRD, CSCR, LRX, and RX methods. Experiments are performed on four datasets and the results demonstrate that our proposed method outperforms the state-of-the-art methods. In the future research, we will try to use the deep learning method to solve for the weight coefficient in order to achieve better results.

Compliance with Ethical Standards

This article does not contain any studies with human participants or animals performed by any of the authors. Ethics Committee approval was obtained from the Institutional Ethics Committee of Nanjing University of Aeronautics and Astronautics to the commencement of the study.

Conflict of Interest   The authors declare that they have no conflict of interest.

References

Bioucas-Dias JM, Figueiredo MAT (2010) Alternating direction algorithms for constrained sparse regression: application to hyperspectral unmixing[C]. Hyperspectral image and signal processing: evolution in remote sensing (WHISPERS), 2010 2nd Workshop on. IEEE, 2010:1–4

Chen Y, NASRABADI NM, TRAN D (2011a) Sparse representation for target detection in hyperspectral imagery [J]. IEEE J Sel Top Sign Proces 5(3):629–640

Chen Y, NASRABADI NM, TRAN TD (2011b) Hyperspectral image classification using dictionary-based sparse representation [J]. IEEE Trans Geosci Remote Sens 49(10):3973–3985

Reed IS, Yu X (1990) Adaptive multiple-band CFAR detection of an optical pattern with unknown spectral distribution [J]. IEEE Trans Acoust Speech Signal Process 38(10):1760–1770

Kong F, Li Y, Guo W (2016) Regularized MSBL algorithm with spatial correlation for sparse hyperspectral unmixing [J]. J Vis Commun Image Represent 40:525–537

Kong F, Guo W, Shen Q et al (2017) Recursive dictionary-based simultaneous orthogonal matching pursuit for sparse unmixing of hyperspectral data [J]. Trans Nanjing Univ Aero Astro 34(4):456–464

Li W, Du Q, Zhang B (2015) Combined sparse and collaboration representation for hyperspectral target detection [J]. Pattern Recogn 48:3904–3916

Li W, Du Q (2015) Collaborative representation for hyperspectral anomaly detection [J]. IEEE Trans Geosci Remote Sens 53(3):1463–1474

Li Y, Xie W, Li H (2017) Hyperspectral image reconstruction by deep convolutional neural network for classification [J]. Pattern Recogn 63:371–383

Li J, Zhao C, Mei F (2010) Detection hyperspectral anomaly by using background residual error data [J]. J Infrared Millim Waves 29(2):150–155

Matteoli S, Diani M, Corsini G (2010) A tutorial overview of anomaly detection in hyperspectral images [J]. IEEE Aerosp Electron Syst Mag 25(7):5–28

MANOLAKIS D, SHAW G (2002) Detection algorithms for hyperspectral imaging applications [J]. IEEE Signal Process Mag 19(1):29–43

Qiu K, Dogandzic A (2010) Variance-component based sparse signal reconstruction and model selection [J]. IEEE Trans Signal Process 58(6):2935–2951

Rakotomamonjy A (2011) Surveying and comparing simultaneous sparse approximation (or group-lasso) algorithms [J]. Signal Process 91(7):1505–1526

Tan K, Du P (2008) Hyperspectral remote sensing image classification based on support vector machine [J]. Infrared Millim Waves 27(2):123–128

Themelis KE, Rontogiannis A, Koutroumbas KD (2012) A novel hierarchical Bayesian approach for sparse semisupervised hyperspectral unmixing [J]. IEEE Trans Signal Process 60(2):585–599

Wipf DP, Rao B (2004) Sparse Bayesian learning for basis selection [J]. IEEE Trans Signal Process 52(8):2153–2164

Wipf DP (2006) Bayesian methods for finding sparse representations[D]. [Ph. D. dissertation], UC, San Diego

Wipf DP, Rao BD (2007) An empirical Bayesian strategy for solving the simultaneous sparse approximation problem [J]. IEEE Trans Signal Process 55(7):3704–3716

Wang L, Wei F, Liu D, Wang Q (2013) Fast implementation of maximum simplex volume-based endmember extraction in original hyperspectral data space [J]. IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing 6(2):516–521

Xie W, Jiang T, Li Y et al (2019a) Structure tensor and guided filtering-based algorithm for hyperspectral anomaly detection [J]. IEEE Trans Geosci Remote Sens 2019:1–13

Xie W, Shi Y, Li Y, Jia X, Lei J (2019b) High-quality spectral-spatial reconstruction using saliency detection and deep feature enhancement [J]. Pattern Recogn 88:139–152

Zhang Y, Du B, Zhang Y et al (2017) Spatially adaptive sparse representation for target detection in hyperspectral images [J]. IEEE Geosci Remote Sens Lett 14(11):1923–1927

Zhao C, Li J, Mei F (2010) A kernel weighted RX algorithm for anomaly detection in hyperspectral imagery [J]. J Infrared Millim Waves 29(5):378–382

Zhao C, Cheng B, Yang W (2012) Algorithm for hyperspectral unmixing using constrained nonnegative matrix factorization [J]. Journal of Harbin Engineering University 3(33):377–382

Zhang Z, Rao B (2011) Iterative reweighted algorithms for sparse signal recovery with temporally correlated source vectors[C]. The 36th International Conference on Acoustics, Speech, and Signal Processing (ICASSP), Brisbane, Australia, 2011: 3932–3935

Zhang Z, Rao B (2012) Recovery of block sparse signals using the framework of block sparse Bayesian learning[C]. The 37th IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), La Jolla, CA, USA, 2012: 3345–3348

Zhang Z, Rao B (2013) Extension of SBL algorithms for the recovery of block sparse signals with intra-block correlation [J]. IEEE Trans Signal Process 61(8):2009–2015