IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received June 7, 2018, accepted August 14, 2018, date of publication August 17, 2018, date of current version September 7, 2018.

Digital Object Identifier 10.1109/ACCESS.2018.2865963

A Hyperspectral Target Detection Framework With Subtraction Pixel Pair Features

JINMING DU[1] AND ZHIYONG LI[2]

[1]School of Electronic Science, National University of Defense Technology, Changsha 410073, China
[2]Hunan Shenfan Technology Co., Ltd., Changsha 410205, China

Corresponding author: Zhiyong Li (Lzylmz75@foxmail.com)

This work was supported by the National Natural Science Foundation of China under Grant 40901216.

ABSTRACT In recent years, due to its strong nonlinear mapping and research capacities, the convolutional neural network (CNN) has been widely used in the field of hyperspectral image (HSI) processing. Recently, pixel pair features (PPFs) and spatial PPFs (SPPFs) for HSI classification have served as the new tools for feature extraction. In this paper, on top of PPF, improved subtraction pixel pair features (subtraction-PPFs) are applied for HSI target detection. Unlike original PPF and SPPF, the subtraction-PPF considers target classes to afford the CNN, a target detection function. Using subtraction-PPF, a sufficiently large number of samples are obtained to ensure the excellent performance of the multilayer CNN. For a testing pixel, the input of the trained CNN is the spectral difference between the central pixel and its adjacent pixels. When a test pixel belongs to the target, the output score will be close to the target label. To verify the effectiveness of the proposed method, aircrafts and vehicles are used as targets of interest, while another 27 objects are chosen as background classes (e.g., vegetation and runways). Our experimental results on four images indicate that the proposed detector outperforms classic hyperspectral target detection algorithms.

INDEX TERMS Target detection, hyperspectral imagery, deep learning, convolutional neural network, subtraction pixel pair features.

I. INTRODUCTION

Hyperspectral technologies are becoming a focus of remote sensing domains in various countries [1]. Due to their high spectral resolution, pixels can be used to recognize small targets of interest [2]. Object detection and recognition via HSI is a central research field with practical applications. In general, detection algorithms can be divided into two categories: supervised and unsupervised. The unsupervised algorithm, which is also called the ''anomaly detection'' algorithm, does not require the use of prior spectral information, reflectance spectra and atmospheric compensation. The supervised algorithm or ''target detection'' algorithm requires prior spectral information of the target of interest [3]. The algorithm proposed in this paper focuses on supervised target detection from HSIs.

At present, several detection algorithms have been put forward from different perspectives. The following four principles characterize methods of hyperspectral target detection.

1) Classic detection algorithms based on multivariate statistical analysis methods and signal processing [1], [4]. These algorithms project spectral features of image data onto a certain plane according to a certain criterion. This ensures that the target and background are located in different positions of the plane. Then, the target is separated from the background by threshold segmentation [4]. A well-known signature-based approach is that of constrained energy minimization (CEM). This algorithm can minimize energy output under constraints of the target signal to ensure target detection [5]. The orthogonal subspace projection (OSP) is a very good target detection operator that determines both the target spectrum of interest and the background spectrum [6], [7]. The above two algorithms are classic unstructured algorithms. Rather than representing background information based on end member signals and corresponding components, structured detectors such as the generalized likelihood ratio test (GLRT), adaptive cosine estimator (ACE), adaptive matched filter (AMF), adaptive subspace detector (ASD) and matched subspace detector (MSD) regard the background as a statistical model. The model conforms to the multivariate Gaussian distribution, in which both the background and noise are modeled as the background [4].
2) Detection algorithms using the kernel function. To better distinguish the target from the background, the kernel function is introduced into traditional linear detection algorithms, forming some new detectors

(e.g., kernel orthogonal subspace projections (KOSP), kernel-based constrained energy minimization (KCEM), kernel matched subspace detectors (KMSD) and kernel-based target constrained interference-minimized filters (KTCIMF)). By mapping data from low dimensional input space to high dimensional feature space via the kernel strategy, nonlinear problems are converted into linear problems [4].

3) Detection algorithms based on sparse representation. The basic premise of introducing sparse theory into HSI target detection algorithms is to represent the original hyperspectral signal as the product of an over-complete dictionary and of coefficients. In general, the sparsest set of coefficients is to be determine such that essential features of the signal can be represented by a few large coefficients [8]. In [9], a joint sparsity model is proposed to determine sparse representations of neighboring pixels. In the algorithm, pixels are decomposed over the given dictionary consisting of training samples of the target and background classes.
4) In recent years, with the continuous emergence of new algorithms in the field of statistical pattern recognition and machine learning, some data-driven target detection methods such as the non-linear manifold learning method [10], transfer learning method [11], background self-learning method [12] and regularization method [13] have been used to process remote sensing image. In particular, a tensor-matched subspace detector (TMSD) is proposed in [14]. In the algorithm, to jointly utilize information of multidimensional hyperspectral data, data are represented as a third-order tensor. This method uses spectral-spatial information and achieves good results.

Although some progress has been made in research on hyperspectral target detection, some challenges remain in its development.

1) Uncertainties in the spectra of targets [4]. Due to varying factors, field measurement reflectance spectral data of a target are not uniquely determined and can vary [4]. This is precisely due to the fact that such uncertainties of the target to be measured cannot be described by a single spectral curve. As is shown in Fig. 1, although the shape of spectral curves of pixels belonging to the focus of this work is approximately the same, the spectral value varies dramatically, complicating accurate target detection and identification.
2) Problems related to mixed pixels [4]. Due to limitations on spatial resolutions and the complexities of ground object distributions, in most cases, one pixel may cover hundreds of square meters with various ground objects and becomes a mixed pixel [4]. Thus, the target object often occupies part of the pixel area and is mixed with a variety of other objects into a single pixel. The mixed pixel problem complicates the identification and recognition of materials. This problem must be addressed urgently for target detection from hyperspectral imagery.
3) The nonlinear problem [1], [4]. At present, most detectors are based on the spectral linear mixed model or pure point model, while high-order feature information for images is not fully utilized. Meanwhile, with an increase in the number of nonlinear factors involved, nonlinear feature distribution will seriously affect the data analysis process, causing traditional methods to not directly apply. Non-linear information often plays an important role in HSI detection and classification.

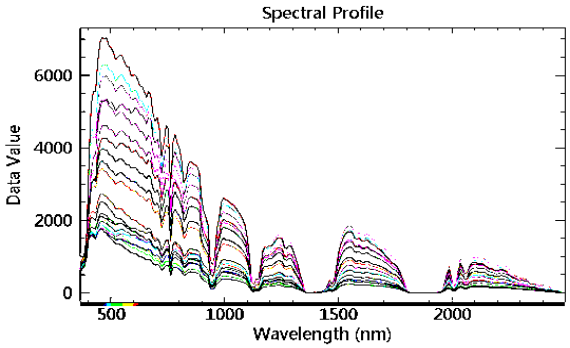


FIGURE 1. Spectral curves of pixels that belong to target of interest in the experimental AVIRIS data.

Recently, deep learning-based methods have drawn increasing attention in the field of hyperspectral image analysis. A variety of neural networks such as the CNN, Deep Belief Network (DBN) and Stacked Auto Encoder (SAE) have been introduced into the realm of hyperspectral data processing and especially in terms of classification. In [16], Convolutional Recurrent Neural Networks (CRNN) are used. Several convolutional layers are used to extract mid-level and locally invariant features from the input data, and then, the following recurrent layers are used to further extract spectrally contextual information from features generated by the convolutional layers.

Recently, novel pixel pair features (PPF) have been proposed as a means to significantly increase the number of training samples involved in [17]. PPF ensure that advantages of the CNN for HSI classification can be actually realized. In addition to PPF, a revamped spatial pixel pair feature (SPPF) is proposed to better exploit spatial/contextual and spectral information [18]. In particular, deep CNN with PPF are applied to hyperspectral anomaly detection for the first time in [20]. In [20] reference data with labeled samples are required for the training procedure, and then the difference between pixels pairs is determined for anomaly detection. Experimental results demonstrate that the method outperforms both classic and state-of-the-art representation-based detectors.

According to the above analyses, the CNN-based target detection framework is robust, as in applying deep learning to hyperspectral remote sensing data processing, the multi-layer expression of the potential distribution of an

HSI can be determined through the deep neural network, and nonlinear features can be extracted. In relying on their powerful learning abilities, deep neural networks can effectively combine feature extraction capacities with classification and recognition features, supporting target detection or terrain classification. Thus, the CNN-based method is helpful in addressing the challenges mentioned above.

Inspired by [17]–[20], a hyperspectral target detection framework with subtraction pixel pair features is presented. First, the size of training samples is enlarged via subtraction-PPF. The training samples can be enlarged by removing any two pixels from background and target classes. Second, by coding the new samples removed between pixels from target classes and other background classes as 1 and new samples removed between pixels of both different and the same background classes as 0, target detection becomes the classification mode of new samples, and the CNN is trained by millions of new samples generated by subtraction-PPF. Finally, for each testing pixel, the input of the trained CNN is the difference between the central pixel and its neighboring pixels, and outputs are the scores of central pixels between 0 and 1. When the testing pixel belongs to target of interest classes, the average score should be close to 1.

This work makes two main contributions:

1) A subtraction pixel pair feature based on the target and background classes is used to obtain a sufficiently large number of samples for the training of the proposed target detection framework, ensuring that advantages of the CNN can be realized.
2) The CNN is used to extract advanced features for target detection from an HSI. All of our experimental results demonstrate that the CNN-based detector with favorable nonlinear mapping and excellent learning capabilities is superior in executing hyperspectral target detection.

The rest of this paper is organized as follows. Section II describes the PPF, SPPF and the proposed subtraction-PPF. The detailed architecture of the CNN and its overall flow is introduced in Section III. By comparing the proposed method with other detectors, experiments are conducted in Section IV, which includes the descriptions of the AVIRIS data, parameters setup, the results and corresponding discussion. Finally, section V summarizes the key ideas in this paper.

II. SUBTRACTION-PPF

To form the CNN, two problems must be solved: a large number of training samples must be obtained, and the CNN must be made a target detection function. Subtraction-PPF is used to address these two problems.

A. PPF AND SPPF

To enlarge samples for the training procedure, PPF are proposed in [17] for HSI classification purposes. As the basic premise of this method, training samples are first paired with any two selected samples based on the following criteria—a pair of samples from the same class is labeled with no change, while that of samples selected from different classes is denoted as 0 [17]. Li *et al.* [20] used a similar approach to expand samples to train the CNN for anomaly detection for an HSI. According to this method, for two types of pixel pairs, one type of sample (the label is denoted as 0) is selected from the same class, while the other type of sample (the label is denoted as 1) is selected from different classes. This was the first way in which a deep CNN was applied for hyperspectral anomaly detection. According to the method, reference data (the Salinas dataset with 16 classes) with ground truthing are required, and the target class is not taken into account. Based on these two approaches, subtraction-PPF values of training samples for the training procedure are put forward. In [18], the SPPF is proposed as a means to better incorporate spatial information, and the geographically co-located pixel selection rule and pair label assignment rule of the SPPF are different from the prior PPF. In the SPPF, only the central pixel and its immediate eight-neighbor pixels are paired, and then, spatial pixel pair features are fed into the designed CNN. The label of the pixel in the SPPF is always coincident with the central pixel, regardless of its neighboring pixels [18]. In [19], a novel feature learning framework, i.e., the simultaneous spectral-spatial feature selection and extraction algorithm, is proposed for hyperspectral image spectral-spatial feature representation and classification.

Taking the CNN presented in this paper for example, the number of parameters in the designed architecture to be trained is approximately 1350000. Traditionally, when the CNN is used for target detection from the HSI, the input of the neural network will be different classes of samples and corresponding labels, and the output of each tested pixel will be its corresponding label. When the tested pixel belongs to the target of interest, the output label will be consistent with the target class or consistent with different background classes. Under such conditions, target detection can be considered a form of classification. However, in reality, there are not this many samples and particularly target samples of interest, and it is essential to the effective functioning of the CNN that there are enough training samples. Taking AVIRIS data used in Salinas [20] as an example, there are 16 classes, and a maximum of 900 samples can be obtained for each class. The total number of available samples of 14400 is much lower than 1350000. Most objects included in the Salinas image are crops and vegetation, which are unsuitable for images containing more manmade objects such as runways and houses.

B. SUBTRACTION-PPF

Inspired by the prior PPF and SPPF, the subtraction-PPF is used to enlarge samples for the training stage.

First, 27 (denoted as k) background classes (denoted as X_b), such as roads, vegetation and tarmac and an aircraft-vehicles target class (denoted as X_t) are manually chosen for several AVIRIS date[1] X values,

[1] https://gulfoilspill.jpl.nasa.gov/cgi-bin/search.pl#

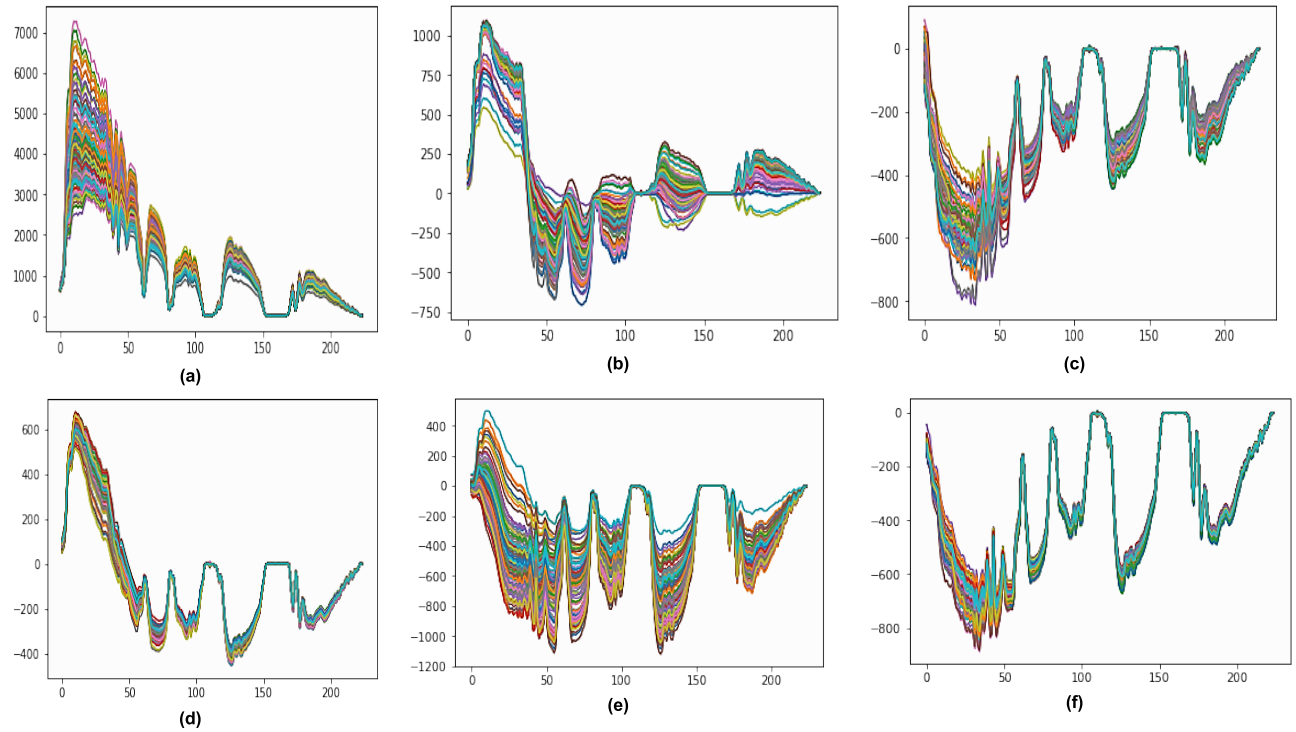


FIGURE 2. Spectral curves of pixels that belong to target of interest in original data and the new data set, the horizontal axis represents the number of bands and vertical axis represents the spectral values for each band. (a) Spectral curves of some targets in original data; (b), (c), (d), (e), (f) Spectral curves of some targets in the new data set.

where $= \{x_i\}$, i $\epsilon\{1, 2 \ldots 27\}$ and where x_i is one of the background classes. By comparing the spectral curves of the aircrafts and vehicles and their appearance in high resolution Google Maps images it is found that the spectra of the aircrafts and vehicles are very similar. Thus, they are selected as target classes to train the proposed CNN, which is used to detect aircrafts and vehicles from AVIRIS images.

Second, to enlarge the samples, a new dataset (denoted X^{new}) is built through target and background subtraction. To generate the pixel pairs, a new pixel (X_{b1}^{new}) generated through the subtraction of any two pixels selected from 27 different classes (x_i, x_j) is coded as 0, the new pixel (X_{b2}^{new}) generated by the subtraction of any two pixels (x_{ii}, x_{jj}) selected from each class (x_i) is coded as 0, and the new pixel (X_t^{new}) created through the subtraction of any two pixels, in which one pixel is drawn from the target class (X_t) while the other is drawn from background classes (X_b), is coded as 1. The new datasets are written as $X^{new} = \{X_{b1}^{new}, X_{b2}^{new}, X_t^{new}\}$, where the background classes (coded as 0) are X_{b1}^{new} and X_{b2}^{new} and the target class (coded as 1) is X_t^{new}. That is,

$$X^{new} = \begin{cases} X_b^{new} \begin{cases} X_{b1}^{new} = x_i - x_j, & i, j \in \{1, 2, \ldots 27, \}, \ i \neq j \\ X_{b2}^{new} = x_{ii} - x_{jj} & x_{ii}, x_{jj} \in \{x_i\}_{i=1}^{27}, \ ii \neq jj \end{cases} \\ X_t^{new} = X_t - x_i, \quad i \in \{1, 2, \ldots 27\} \end{cases} \tag{1}$$

As is shown in (1), the following three types of pixel pairs can be obtained: $x_i - x_j$, $x_{ii} - x_{jj}$, and $X_t - x_i$.

When 300 (denoted as n) samples are selected from each class and when the number of classes is 27 (denoted as k), the total number N1 of X_{b1}^{new} can be calculated from all combinations. That is,

$$N1 = C_k^2 \times n \times n = C_{27}^2 \times 300 \times 300 = 31590000 \tag{2}$$

The total number of X_{b2}^{new} is N2. That is,

$$N2 = C_n^2 \times k = C_{300}^2 \times 27 = 1210950 \tag{3}$$

The total number of X_t^{new} is N3. That is,

$$N3 = n \times n \times k = 300 \times 300 \times 27 = 2430000 \tag{4}$$

From the above measures, the new dataset X^{new} is built and is divided into two classes: the background class (denoted 0) with number n0 and the target class (denoted 1) with number n1. The label and number of samples of each class are written as

$$Label(X^{new}) = \begin{cases} 0, & n0 = N1 + N2 = 32800950 \\ 1, & n1 = N3 = 2430000, \end{cases} \tag{5}$$

It is easy to observe that the number of the new datasets is much larger than the number of parameters used in the proposed CNN. As is evident from the spectrum curves (as shown in Fig. 2 and Fig. 3) of several samples of the new dataset X^{new}, X^{new} is very different from the original AVIRIS

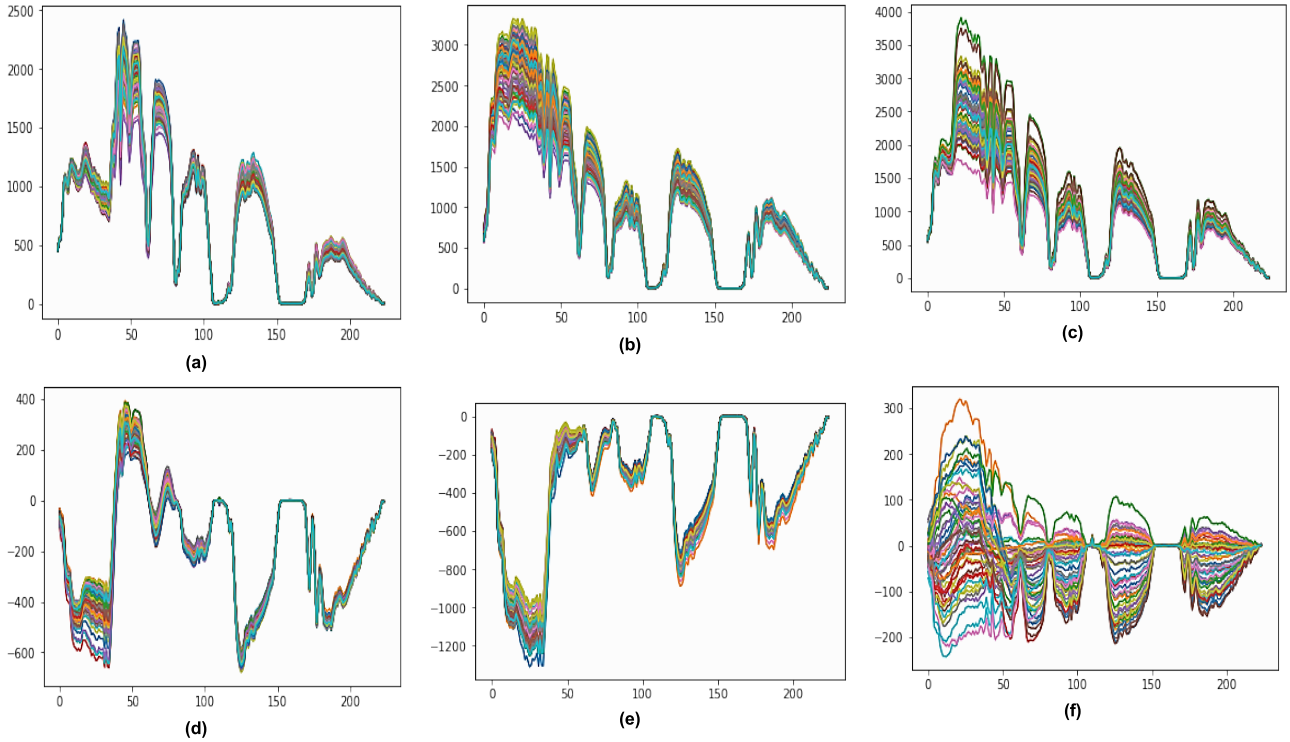


FIGURE 3. Spectral curves of pixels that belong to background in original data and the new data set, the horizontal axis represents the number of bands and vertical axis represents the spectral values for each band. (a), (b), (c) Spectral curves of some backgrounds in original data; (d), (e), (f) Spectral curves of some backgrounds in the new data set.

data X value. In the subtraction-PPF, both target and background classes of X^{new} contain different spectra because most samples of the new dataset are created from a combination of any two different samples. Compared to the spectra value of samples in X, which is greater than zero, the spectra value of samples in X^{new} can be less than zero due to the presence of numerical values, causing the spectrum to be more diverse and easier to distinguish. However, it is difficult to identify new spectra from traditional methods, as the spectrum in X^{new} and the spectrum of typical objects in X are quite different. Thus, to identify the target spectrum of so many different new spectra, a non-linear detector with strong recognition capacities must be used. At this point, the target detection problem defined in X is transformed into the classification problem defined in X^{new}. In the new dataset, the target class is only one of many categories used. The difference observed is attributed to the fact that the labels of other classes are denoted as the same. For example, for class X_{b1}^{new}, which is created from the subtraction of any two pixels selected from 27 different classes, according to formula (2), the number of new classes is C_{27}^2, but they are denoted as 0. This means that these C_{27}^2 classes are merged into the background class of the new dataset. Similarly, according to formula (4), 27 classes are merged into the target class of the new dataset. This increases the diversity and sample number of the target class. Like other classification problems, the CNN is used to classify new classes in the new dataset. Thus, the transformation can be used to address class imbalances.

In experiments, only 150 samples for each background class and 300 samples for the target class are manually chosen for several original AVIRIS dates. Then these samples are paired to generate the new dataset. According to the above subtraction method and according to formulas (1) to (5), 8199225 background samples and 2430000 target samples in the new dataset can be obtained as training samples.

C. COMPARISON OF PPF, SPPF, AND SUBTRACTION-PPF

Compared to the PPF and SPPF, the proposed subtraction-PPF takes a target class into account. PPF features are pixels generated from paired pixels on reference data with labeled samples. In [17], [18], and [20], pixel pair features are utilized for HSI classification and anomaly detection. In PPF [20], paired pixels are selected from the Salinas dataset with sixteen classes, and the same pairing method is applied to all sixteen classes. In this paper, however, aircrafts and vehicles are regarded as the target class, and target features of the subtraction-PPF are generated from paired pixels between the target class and other background classes. Thus, the subtraction-PPF ensures that the CNN is a target detection function.

In addition, to better incorporate spatial information, the SPPF only selects the central pixel and its eight-neighbor pixels at the training and testing stage [18]. In the subtraction-PPF, spatial information is not considered in the training stage, as paired pixels used in the subtraction stage are chosen randomly. However, spatial information is used in

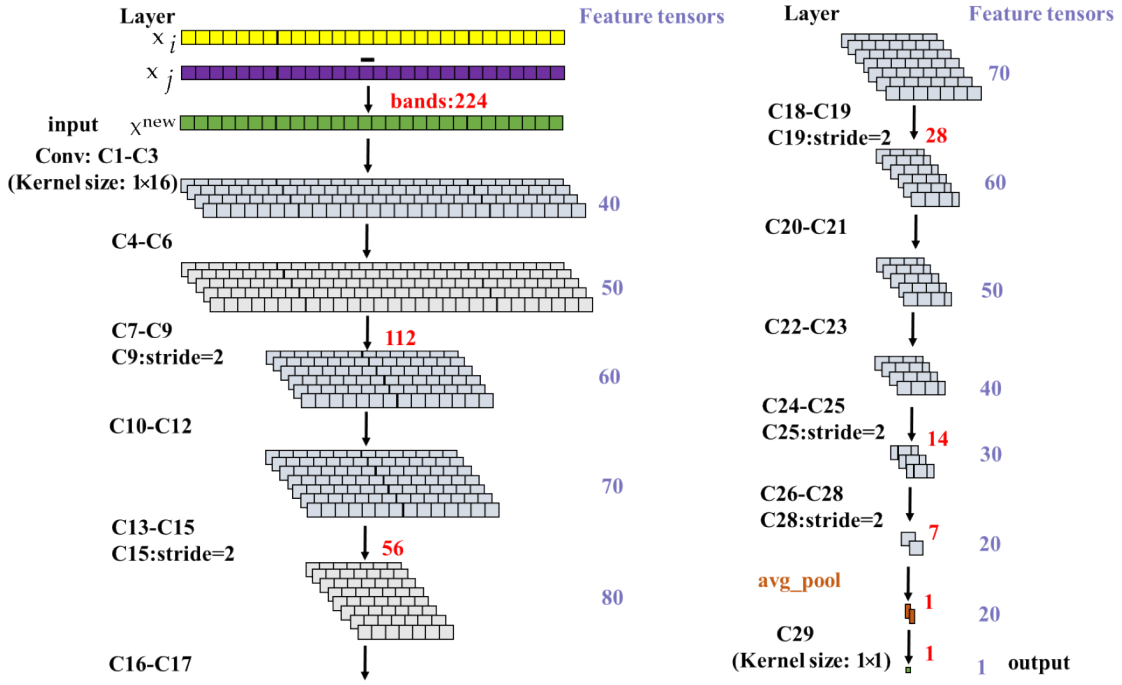


FIGURE 4. Architecture of the CNN-based target detection framework.

the testing stage, and to manage edge pixels, more neighboring pixels are paired with the testing pixel. It is conducive to reduce the effects of edge pixels. Given that the sizes of targets in an image vary, different processing windows are used when a testing pixel is paired with other neighboring pixels, and it is found that the larger the processing window is, the more pixels are paired with the central pixel. This renders the contrast of spatial information between the testing pixel and neighboring pixels complementary to the practical distribution of ground objects.

Moreover, the subtraction-PPF offers abundant characteristic information. In PPF [17], [20], reference data have sixteen classes, and most objects include crops and vegetation. Thus, PPF features [17], [20] are not suitable for the detection of targets in images with artificial objects. However, the subtraction-PPF described in this paper is generated from twenty-eight different ground objects that are manually chosen from several AVIRIS dates. This renders the subtraction-PPF more suited for target detection.

III. CNN-BASED TARGET DETECTION FRAMEWORK

A. ARCHITECTURE OF THE PROPOSED CNN

After constructing the new dataset, the designed CNN can be trained. The architecture of the CNN is shown in Fig. 4. The design of the neural network structure described in this paper is inspired by CNNs described in [17] and [20]. The CNN described in this paper contains more layers and performs different functions. The designed CNN framework contains twenty-nine convolutional layers and one average-pooling layer. Linear unit layers rectified after each convolutional layer are used to accelerate the convergence of the stochastic gradient descent algorithm and to ensure that the trained network exhibits moderate sparsity. It is worth noting that convolutional layers with stride two (C9, C15, C19, C25, and C28) are used to limit spectral dimensionality.

As is illustrated in Fig. 4, the input of the CNN is a $1 \times 224 \times 1$ tensor and a corresponding 0 or 1 label. The input tensor is converted by the 1×224 spectral vector of the new dataset X^{new}. Then, the first convolutional layer (C1) primarily filters the input $1 \times 224 \times 1$ tensor with forty $1 \times 16 \times 1$ kernels, producing a $1 \times 224 \times 40$ tensor. Then, the second convolutional layer (C2) filters the input $1 \times 224 \times 40$ tensor with forty $1 \times 16 \times 1$ kernels, producing a $1 \times 224 \times 40$ tensor. Then, the third convolutional layer (C3) filters the input $1 \times 224 \times 40$ tensor with forty $1 \times 16 \times 1$ kernels, producing a $1 \times 224 \times 40$ tensor. From C1 to C3 (C1-C3 in Fig. 4), nonlinear features are extracted. In short, to obtain high-level features, the networks must use more convolutional layers, and thus, 28 convolutional layers (from C1 to C28) are continuously designed.

Like layers from C1 to C3, the output of C6 is fifty 1×224 feature tensors, which are obtained by applying fifty $1 \times 16 \times 1$ kernels to the $1 \times 224 \times 50$ tensor generated by C4. From C1 to C8, the output of each layer has the same dimensions as the original input vector because the stride is valued at one. When each layer enters C9 with stride two, the output is sixty 1×112 feature tensors such that the spectral dimension is reduced by half. Following the same convolution operation as that described above, the output of the twenty-eighth convolution layer (C28) is twenty

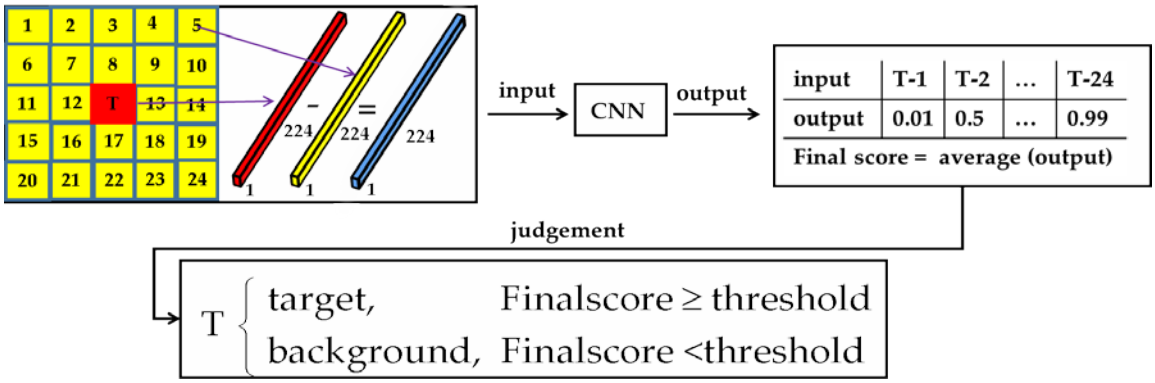
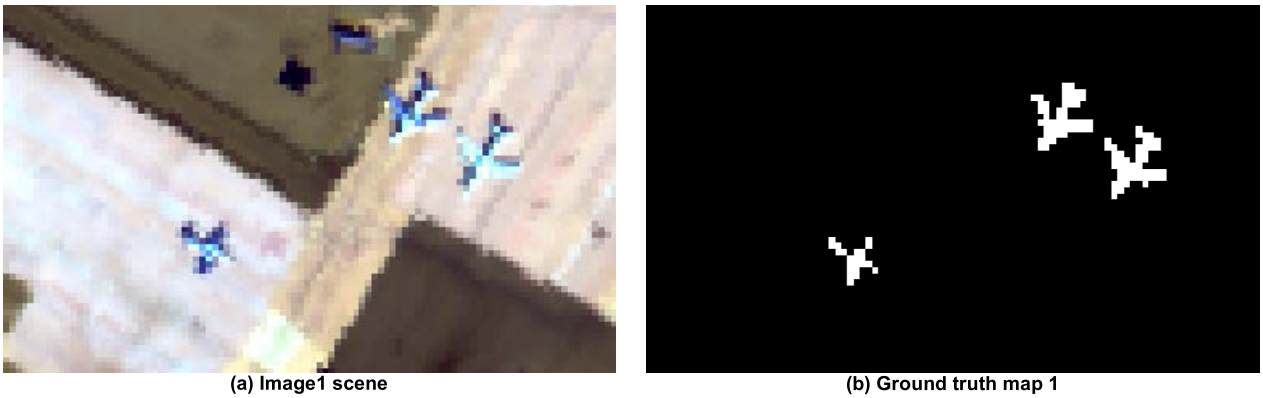


FIGURE 5. Flowchart of the CNN-based target detection in HSI with single window.



(a) Image1 scene (b) Ground truth map 1

FIGURE 6. Pseudo color image1 scene using bands 185, 130 and 35. (b) Ground truth map of image1 with 123 target pixels.

1×7 feature tensors. After 28 convolutional layers are applied, another convolutional layer (C29) filters a 1×20 tensor, which is generated by an average-pool layer applied to the output of C28 with a $1 \times 16 \times 1$ kernel, producing the final 1×1 output score of between 0 and 1.

B. FLOWCHART OF CNN-BASED TARGET DETECTION VIA THE HSI

When testing, the same subtraction method is used to generate a new input vector for the testing pixel. As shown in Fig. 5, the input of the trained CNN is a new 1×224 vector, which is generated through subtraction between the central testing pixel and neighboring pixel. When using twenty-four neighborhood pixels, the output is twenty-four scores of between 0 and 1. Then, the mean of the twenty-four scores is compared to the prescribed threshold; when the mean of the twenty-four scores $>$ the threshold, the testing pixel T belongs to the target; otherwise, it belongs to the background.

Generally, the threshold is set to 0.5 because when a testing pixel belongs to the target of interest classes, the final score should be close to 1, and otherwise, it is close to 0. For ROC generation, the threshold is gradually changed from 0 to 1, and thousands of thresholds between the minimum and maximum of the detection output map are used to calculate the ROC curve.

IV. EXPERIMENT AND DISCUSSION

In the experiment, only one CNN model is trained and used for all images.

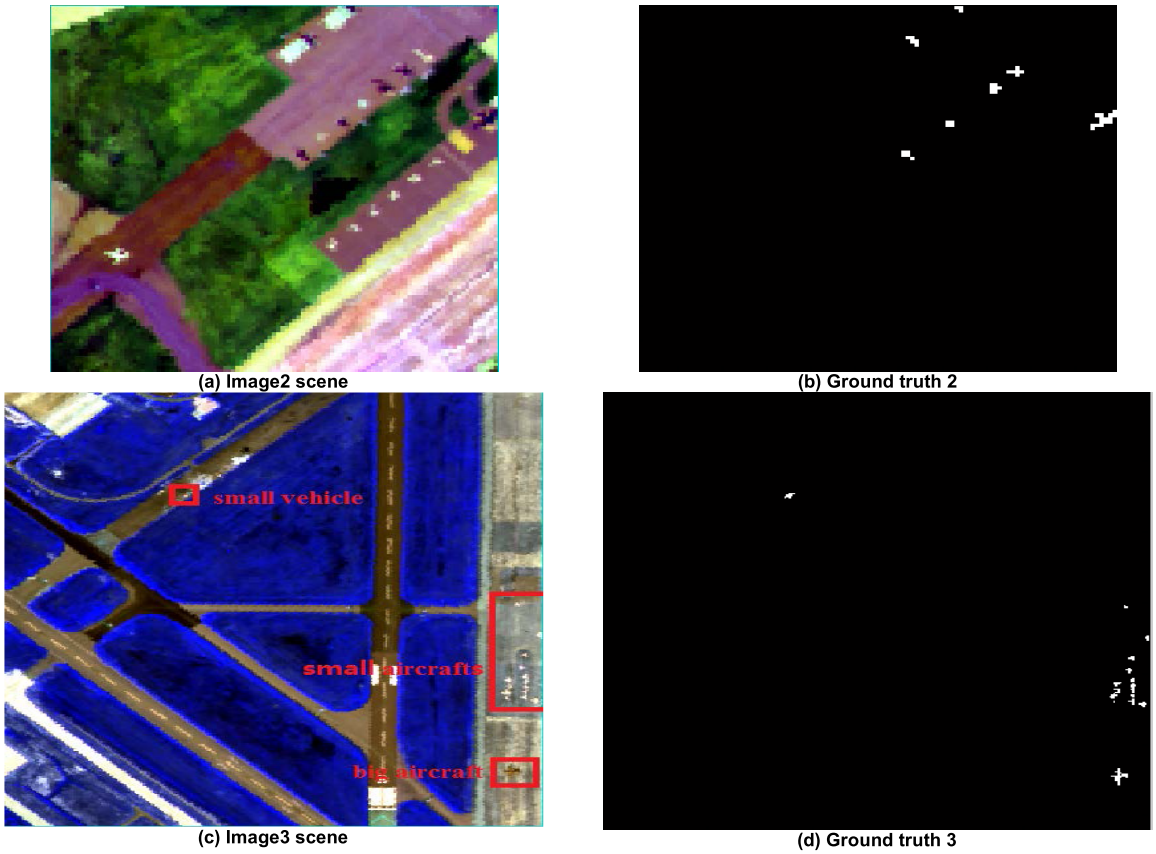
A. DATASET AND ROC

1) DATASET

To demonstrate the performance of the CNN-based target detection framework, four images drawn from three AVIRIS datasets are employed. All images with a 3.5 m spatial resolution are gathered by the Airborne Visible Infrared Imaging Spectrometer [21] sensor and have 224 spectral channels in wavelengths ranging from 370 to 2510 nm, in which the wavelengths of 1350–1420 and 1810–1940 nm are water-absorption bands. All 224 bands are used in the experiments.

The first image (image1) was collected by the Airborne Visible Infrared Imaging Spectrometer sensor from San Diego Airport, CA, USA [22]. This scene consists of 60×100 pixels and there are three planes in the image, which consist of 123 pixels. The image1 scene and the ground truth map of image1 are shown in Fig. 6.

The second and third images (image2 and image3) were collected by the Airborne Visible Infrared Imaging Spectrometer sensor. These two images come from the same data



(a) Image2 scene (b) Ground truth 2

(c) Image3 scene (d) Ground truth 3

FIGURE 7. (a) Pseudo color image2 scene using bands 180, 128 and 30. (b) Ground truth map of image2 with 45 target pixels. (c) Pseudo color image3 scene using bands 16, 25 and 50. (d) Ground truth map of image3 with 120 target pixels.

covering Mississippi, USA. The image2 consist of 109 × 109 pixels and there are seven aircrafts in the image, which consist of 45 pixels. The image3 consist of 311 × 259 pixels and there are fifteen aircrafts and one vehicle in the image, which consist of 120 pixels. The image scenes and the ground truth maps of image2 and image3 are shown in Fig. 7.

The fourth image (image4) was collected by the Airborne Visible Infrared Imaging Spectrometer sensor. This image is the AVIRIS data covering a parking lot on Galveston Island, Texas, USA. This scene consists of 70 × 210 pixels and there are some vehicles in the image, which consist of 330 pixels. The image scene and the ground truth map of image4 are shown in Fig. 8.

2) ROC

The receiver operating characteristic (ROC) curve, which shows the tradeoff between probability of false alarm (denoted as Pf) and probability of detection (denoted as Pd), is used to evaluate the experiment results. In addition, area under the curve (AUC) is calculated to measure the performance of the ROC, the larger the AUC, the better the performance of the detector [23]. To calculate the ROC curve, the detection map is first normalized to (0, 1). After picking thousands of thresholds between the minimum and maximum of the detection output map, the resulting Pf (which means that background pixels are detected as target) and Pd (which means that target pixels are detected as target) are plotted by comparing with the ground truth map [9], [24]. Pd is the ratio of the number of detected true target pixels in the detection map to the number of true target pixels in the ground truth map; Pf is the ratio of the number of detected false pixels in the detection map to the total number of pixels in the entire tested image. It must be noted that the targets pixels in the ground truth maps are artificially marked one by one relying on experience and knowledge of the spectral information. In addition, high resolution images of Google Maps are also used as the reference to make ground truth maps.

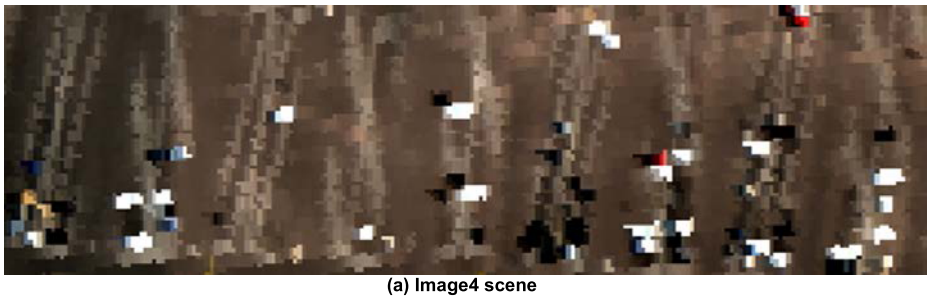
B. PARAMETER SETUP

To maximize the effectiveness of the neural networks, appropriate parameters must be applied, i.e., window sizes, convolution kernel sizes, learning rates and classes of samples.

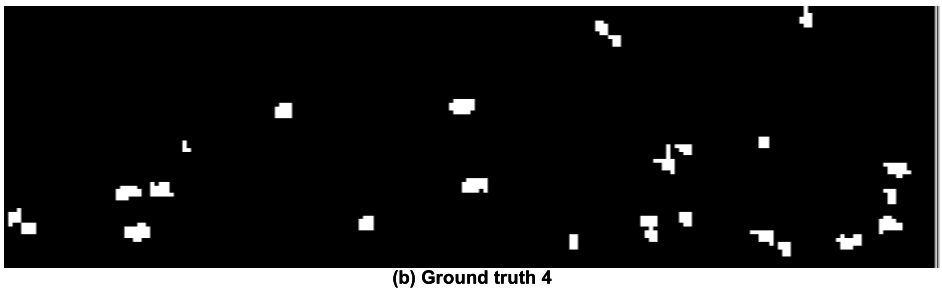
1) WINDOW SIZE

As numerous experiments show that CNN window sizes have a strong effect on target detection results, this is discussed in detail in this paper.

Windows of different sizes are used to obtain input data in the experiments (as shown in Fig. 5). Tables 1 to 4 demonstrate the effects of windows of various sizes and of execution



(a) Image4 scene



(b) Ground truth 4

FIGURE 8. (a) Pseudo color image4 scene using bands 180, 125 and 30. (b) Ground truth map of image4 with 330 target pixels.

TABLE 1. AUC (%) Performance and execution time (seconds) of CNN-based detector for various sizes of single windows using image1.

	Single	Single	Single	Single	Single	Single	Single	Single
Window size	3×3	5×5	7×7	9×9	11×11	13×13	15×15	17×17
auc	0.9804	0.9970	0.9964	0.9763	0.9953	0.9854	0.9840	0.9948
time	33	38	62	88	116	150	199	240

TABLE 2. AUC (%) Performance and execution time (seconds) of CNN-based detector for various sizes of double windows and 4 neighbors using image1.

	Single4	double	double	Double	double	double	double	double
Window size	4	3×5	5×7	7×9	5×11	9×13	5×13	7×15
auc	0.9498	0.7154	0.7484	0.7051	0.7202	0.7293	0.7209	0.713
time	21	29	39	45	98	95	123	156

TABLE 3. AUC (%) Performance and execution time (seconds) of CNN-based detector for various sizes of single windows using image2.

	Single	Single	Single	Single	Single	Single	Single	Single
Window size	3×3	5×5	7×7	9×9	11×11	13×13	15×15	17×17
auc	0.9833	0.9802	0.9811	0.9866	0.9915	0.9934	0.9928	0.9913
time	45	75	120	178	224	297	383	471

TABLE 4. AUC (%) performance and execution time (seconds) of CNN-based detector for various sizes of double windows and 4 neighbors using image2.

	Single4	double	double	double	double	double	double	double
Window size	4	3×5	5×7	7×9	5×11	9×13	5×13	7×15
auc	0.9445	0.2017	0.2168	0.2627	0.2474	0.3120	0.2690	0.6743
time	40	56	75	86	189	181	262	309

times using image1 and image2. When using a single window, central and neighboring pixels are used for subtraction. Taking a 5 × 5 single window as an example, twenty-four input vectors are obtained through subtraction between the central testing pixel and its twenty-four neighboring pixels (as shown in Fig. 5). When using a double window, only pixels between the inner window and outer window participate in the calculation, preventing pixels within the inner window from affecting the calculation. Taking a 3 × 5 single window as an example, sixteen input vectors are obtained by subtraction between the central testing pixel and pixels between the inner and outer windows. From Tables 1 to 4,

TABLE 5. The sizes of window that suitable for four images.

	Image1	Image2	Image3	Image4
Window size	5×5	13×13	21×21	31×31

TABLE 6. AUC (%) performance of CNN-based detector with different learning rate when using image1.

Learning rate	0.5	0.1	0.05	0.01	0.001	0.0001
auc	Non-convergence	Non-convergence	0.6221	0.8233	0.9970	0.9010

TABLE 7. AUC (%) performance Of CNN-based detector with different convolution kernel size when using four images.

	1×3	1×7	1×11	1×16
Image1	0.8826	0.8310	0.9406	0.9970
Image2	0.9745	0.9912	0.9870	0.9933
Image3	0.9566	0.9936	0.9911	0.9928
Image4	0.9835	0.9213	0.9547	0.9898

it can be clearly observed that a single window is better than a double window, as the sizes of targets on the image vary. An inner window of a specific size may only be suitable for detecting targets of the corresponding size while being unsuitable for detecting targets of different sizes. Therefore, although double windows protect pixels within the inner window and save time, detection capacities are poor. As is shown in Tables 2 and 4, single windows and even the four neighboring windows are more suited to detecting targets of different sizes than double windows. The single window can adequately reflect the difference between testing and surrounding pixels. Our experiments of image3 and image4 also show that better detection performance can be achieved when using a single window. For different data, however, different situations are observed, as the larger the window is, the higher the precision. It is worth noting that the larger the window is, the longer the calculation time, and so it is necessary to use the right window according to actual conditions. Sliding windows of different sizes are used to obtain a different number of input vectors for each image and the most suitable windows for each image are shown in Table 5.

2) CONVOLUTION KERNEL SIZE

During deep feature extraction using the CNN, it is important to address the configuration of the designed neural network. Convolution kernel sizes recommended in recent studies for the CNN framework are 5×5, 7×7 or 9×9 [25]. Unlike two-dimensional images, inputs of the CNN examined in this paper are one-dimensional vectors. Therefore, four kinds of convolution kernels of different sizes were applied in the experiment, i.e., 1×3, 1×7, 1×11 and 1×16. The kernel size is important because it dictates the size of the feature to extract. Compared to features extracted from every three neighboring bands when using kernels of 1×3, features extracted from every sixteen neighbor bands when using kernels of 1×16 incorporate more information. This better reflects connectivity between multiple bands. Although a large kernel can mix more information than a small kernel, this takes more time. Table 7 shows the AUC performance of the proposed CNN-detector with different convolution kernel sizes when using four images. It is evident that the larger the size of the kernel, the greater the AUC value generated through the four experiments. The detector with 1×16 kernels works well when applied to all four images. Based on the computing capacity of the computer used in the experiment, the size of kernels used in the proposed CNN is 1×16.

3) LEARNING RATE

The learning rate determines how fast a parameter moves toward its optimum value. When the learning rate is too fast, it is likely to cross the optimal value, and when the learning rate is too slow, optimization may be too inefficient, causing the networks to not converge over a long period. The learning rate is crucial to the performance of the algorithm, as it determines the convergence speed of backpropagation and can significantly affect training performance [20]. To find a suitable learning rate, various values, i.e., 0.5, 0.1, 0.05, 0.01, and 0.001, are used in the experiment, and a learning rate decay strategy is adopted to balance the training speed and loss. For example, the learning rate is initially set as 0.1 for the decay strategy, and after hundreds of thousands of training steps are completed (i.e., 100000 steps), the value is decreased by multiplying by 0.1. Then, the value becomes 0.01. During the training progress, cross entropy is used as an index to judge whether the network is convergent. Thus, a suitable value can be directly found via the decay strategy. Through experiments, it is found that a fast learning rate (e.g., 0.1) causes the designed CNN to diverge. As shown in Table 6, the most suitable learning rate is measured at 0.001 for the designed CNN.

4) SAMPLE CLASSES AND EXPERIMENTAL ENVIRONMENT

For the first experiment, 13 different types of objects are used for the production of a new dataset, and 28 different types of objects are used in the second and subsequent experiments.

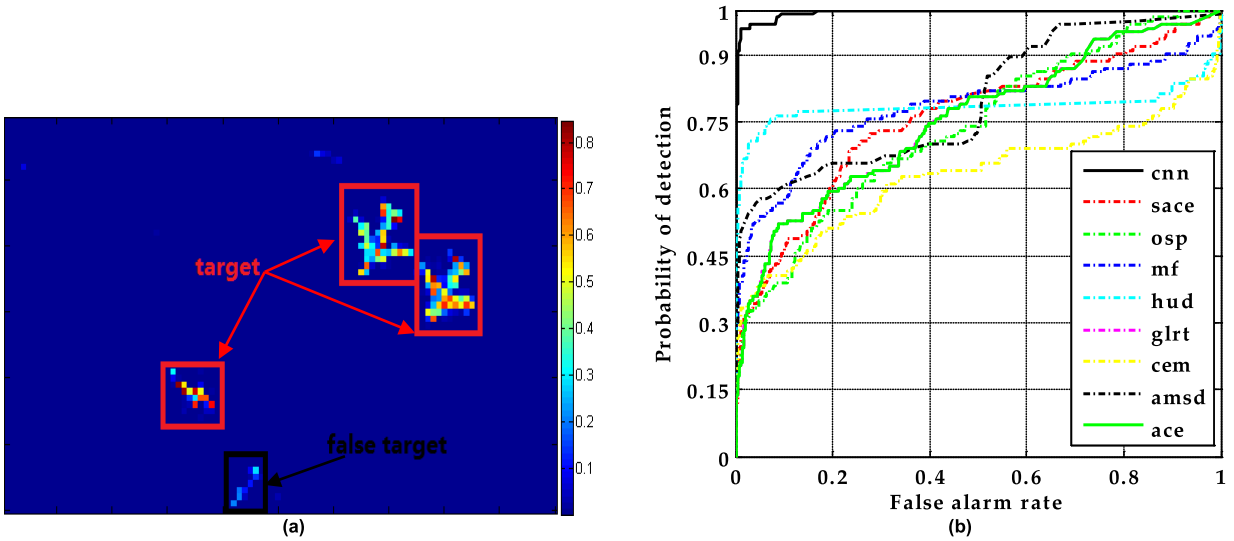


FIGURE 9. (a) Output map of the CNN-based detector using image1. Pixels within the red frame belong to targets of interest. Pixels within the black frame are false targets. For each pixel, the larger the value, the deeper the red color, the greater the possibility that the pixel belongs to the target. (b) ROC curves of different detectors using image1, the horizontal axis represents the probability of false alarm and vertical axis represents the probability of detection.

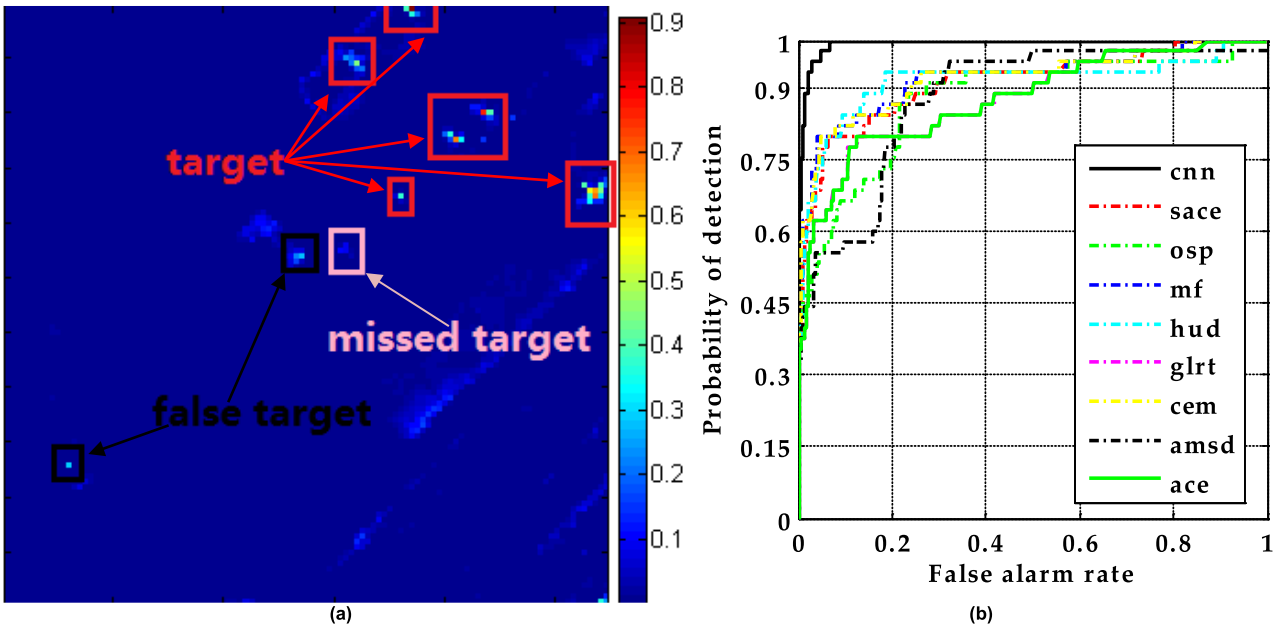


FIGURE 10. (a) Output map of the CNN-based detector when using image2. Pixels within the red frame belong to targets of interest. Pixels within the black frame are false targets. Pixels within the pink frame are missed targets. For each pixel, the larger the value, the deeper the red color, the greater the possibility that the pixel belongs to the target. (b) ROC curves of different detectors using image2, the horizontal axis represents the probability of false alarm and vertical axis represents the probability of detection.

TABLE 8. AUC (%) performance of different detectors using image1.

ACE	AMSD	CEM	GLRT	HUD	MF	OSP	SACE	CNN
0.7532	0.7971	0.6306	0.7537	0.7840	0.7813	0.7368	0.7572	0.9970

The CNN trained by 28 types of objects is more accurate because it can identify more classes. Based on the analysis given in Section II, when the class number is 13, the number of new samples is 7558200, which is less than the number of new samples generated by 28 classes. Thus, 28 classes are used to produce the subtraction-PPF.

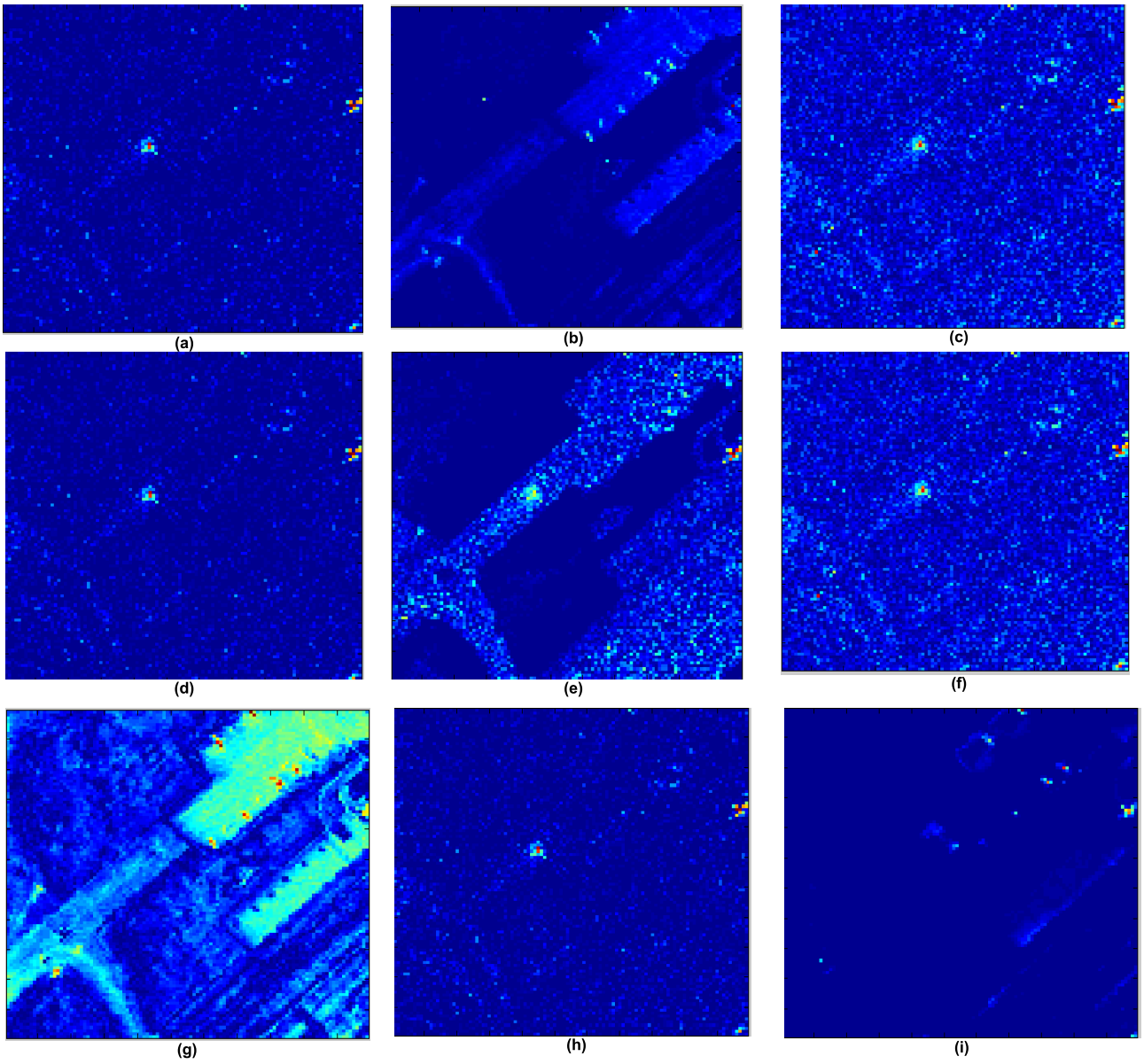


FIGURE 11. Output map of different detectors when using image2. For each pixel, the larger the value, the deeper the red color, the greater the possibility that the pixel belongs to the target. (a) ACE detection map. (b) AMSD detection map. (c) CEM detection map. (d) GLRT detection map. (e) HUD detection map. (f) MF detection map. (g) OSP detection map. (h) SACE detection map. (i) CNN-based detection map.

TABLE 9. AUC (%) performance of different detectors using image2.

ACE	AMSD	CEM	GLRT	HUD	MF	OSP	SACE	CNN
0.8770	0.8786	0.9196	0.8771	0.9145	0.9212	0.8766	0.9118	0.9933

TABLE 10. AUC (%) performance of different detectors using image3.

ACE	AMSD	CEM	GLRT	HUD	MF	OSP	SACE	CNN
0.8704	0.7832	0.8687	0.8336	0.8775	0.8787	0.8273	0.8516	0.9928

All experiments are implemented in the Tensorflow deep learning framework and are executed on an Intel(R) Core(TM) i7-6700 CPU desktop PC with NVIDIA GeForce GTX 1060 (3GB video memory), and 8GB of RAM. The desktop PC operating system is Ubuntu 16.04 and all the programs of the proposed CNN framework are implemented

TABLE 11. AUC (%) performance of different detectors using image4.

ACE	AMSD	CEM	GLRT	HUD	MF	OSP	SACE	CNN
0.8056	0.8542	0.6954	0.8059	0.7301	0.6919	0.7169	0.6882	0.9898

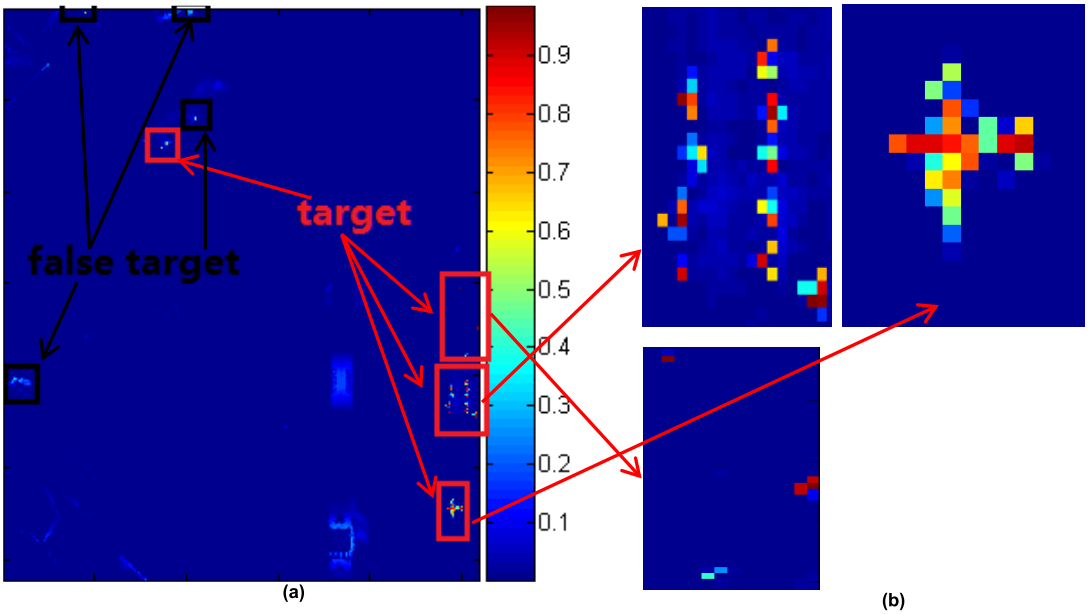


FIGURE 12. (a) Output map of the CNN-based detector when using image3. Pixels with in the red frame belong to targets of interest. Pixels within the black frame are false targets. For each pixel, the larger the value, the deeper the red color, the greater the possibility that the pixel belongs to the target. (b) Enlarged scenes of detected target area.

using Python language. All the classical detectors are carried out on the platform of MATLAB (2014).

C. COMPARISON OF DIFFERENT DETECTORS

To investigate the performance of the proposed CNN-based target detector, the detector is compared to eight classic detectors: the CEM, OSP, ACE, GLRT, MSD, Adaptive Matched Subspace Detector (AMSD), Signed Adaptive Cosine Estimator (SACE), Matched Filter (MF) and Hybrid Unstructured Detector (HUD) [26]–[30]. These programs are carried out using MATLAB. For our comparative analysis, all 224 bands are maintained. For classic detectors, twenty pixels are selected from the aircraft-vehicle class manually applied as shown in Section II. The spectrum of the twenty pixels is fed into the detector as a priori target information, and the best of the twenty test results is taken as the output of each classical detector.

1) EXPERIMENTAL RESULTS OF IMAGE1

There are three big aircrafts in image1. From Fig. 9(a), it can be clearly seen that the shape of detected aircrafts on the detection map. Although there are false target pixels, the values of these pixels are small. The false targets exist because these pixels are at the junction of the two roads. Both the AUC performance (as shown in Table 8) and the ROC curves (as shown in Fig. 9(b)) of different detectors show that the CNN-based target detector outperforms other detectors on image1.

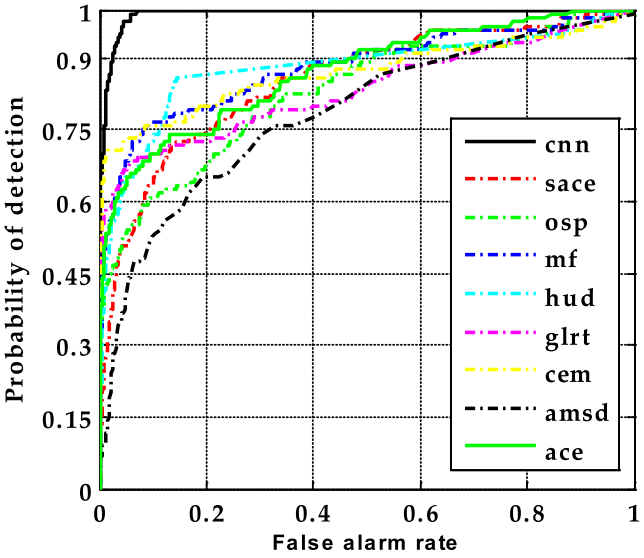


FIGURE 13. ROC curves of different detectors using image3, the horizontal axis represents the probability of false alarm and vertical axis represents the probability of detection.

2) EXPERIMENTAL RESULTS OF IMAGE2

From Fig. 10(a), it can be seen that six targets can be detected. As these targets are small, it is difficult to distinguish their shape. Although some of the targets even have only one or two pixels, they are still detected, proving that the proposed detector has the ability to detect

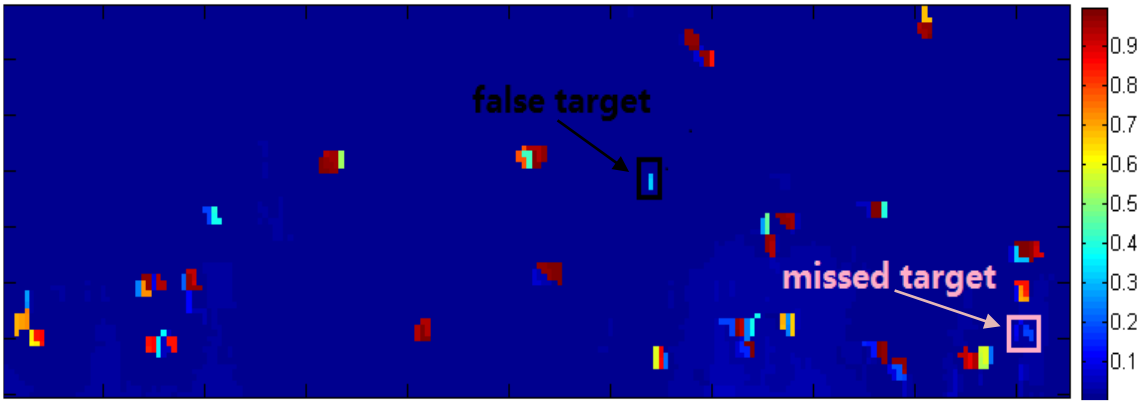


FIGURE 14. Output map of the CNN- based detector when using image4. Pixels within the red frame belong to targets of interest. Pixels within the black frame are false targets. Pixels within the pink frame are missed targets. For each pixel, the larger the value, the deeper the red color, the greater the possibility that the pixel belongs to the target.

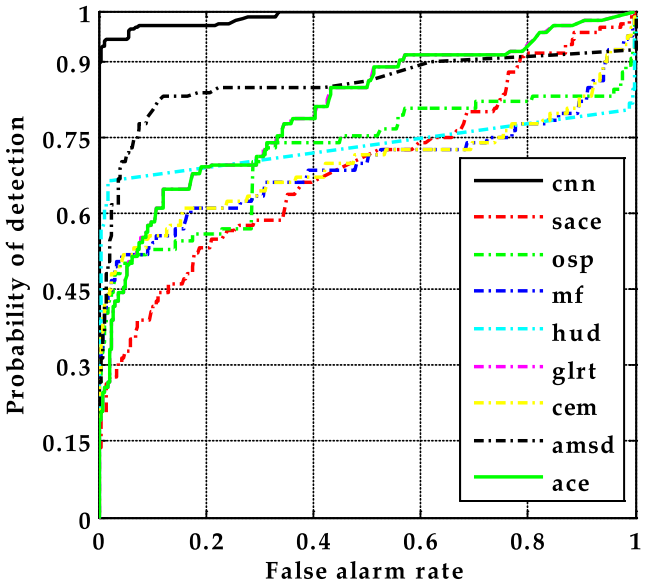


FIGURE 15. ROC curves of different detectors using image4, the horizontal axis represents the probability of false alarm and vertical axis represents the probability of detection.

small targets. But one target is missed and two targets are false. Both the AUC performance (as shown in Table 9) and the ROC curves (as shown in Fig. 10(b)) show that the CNN-based target detector outperforms other detectors on image2. Fig. 11 shows different detection maps of different detectors. It can be seen that other detectors, such as OSP and AMSD, can detect the target, but there are more false targets in background.

3) EXPERIMENTAL RESULTS OF IMAGE3

In image3, there are three different sizes and different kinds of targets. As shown in Fig. 7 (c), there are one big aircraft, fourteen small aircrafts and one small vehicle. Although the shape of spectral curves of three types of targets is approximately same, the spectral value varies dramatically (as shown in Fig. 1). From Fig. 12, it can be seen that all the targets are detected. And there are false targets in the detection map. Both the AUC performance (as shown in Table 10) and the ROC curves (as shown in Fig. 13) show that the CNN-based target detector outperforms other detectors on image3. It also can be seen that the proposed detectors can identify targets with different sizes.

TABLE 12. Execution time (seconds) of CNN-based detector using four images.

	Size	Window size	Time (s)
Image1	60×100	5×5	38
Image2	109×109	13×13	297
Image3	311×259	21×21	4757
Image4	70×210	31×31	1892

4) EXPERIMENTAL RESULTS OF IMAGE4

Image4 contains a number of vehicles with different materials. From Table 11, it can be seen that the detection of the classical detectors is much worse than the CNN-based detector. Because of the limitation of spatial resolution, the detector could not separate each vehicle, but it could detect a few vehicles parked together. Although there are false targets and missed targets on the detection map (as shown in Fig. 14), the AUC performance (as shown in Table 11) and the ROC curves (as shown in Fig. 15) show that the CNN-based target detector outperforms other detectors on image4.

5) TIME CONSUMPTION

The algorithm requires time for sample selection, CNN training and target detection. Sample selection can take some time, but manually selected samples are accurate and reusable. Due to the large number of samples and parameters considered, the training process can take several hours. As shown in Tables 1 to 4 and in Table 12, when the trained CNN detects a target in different images, the execution time is dependent on the sizes of the detected image and sliding window.

In general, the larger the size of the tested image and sliding window, the longer the execution time. It is clear that much time is consumed for image3 and image4, mainly because the suitable sliding window size is large.

D. SUMMARY

For the experimental analysis effects of window sizes, convolution kernel sizes and learning rates were studied. We found that single windows work better than double windows, and a suitable processing window was found for each image. We also found that the sizes of convolution kernels affect detection performance because large kernels may incorporate more information that better reflects connectivity between multiple bands, and the 1×16 kernel is used according to the computer's computation capabilities. After applying different learning rates through the training procedure, the most appropriate learning rate is measured as 0.001.

Experimental results for the four images demonstrate the validity of the proposed detector in three respects. First, with a lower false alarm rate and with higher accuracy, the AUC of the CNN-based detector for four images (up to 98%) is higher than that of the other detectors (less than 93%). Second, the CNN-based detector can not only detect small targets but can also detect targets of different sizes, and the proposed detector works well when applied to the four experimental images. Third, as shown in Fig. 1, although the shapes of spectral curves of the target are approximately the same, the spectral value varies dramatically. These targets can be detected using the proposed CNN-based detector.

V. CONCLUSIONS

In this paper, a CNN-based hyperspectral target detection framework with subtraction pixel pair features is presented. To obtain a sufficiently large number of training samples and render the CNN a target detection function, the subtraction-PPF is used, and the target detection problem of original data is transformed into a classification problem for new data, which is generated through the subtraction-PPF.

The performance of the proposed algorithm is assessed by comparing it to eight classic detectors. The following three advantages of the proposed algorithm are identified.

1) Unlike algorithms that need select good bands in the preprocessing stage, the CNN-based detector identifies poor and water-absorption bands as useful information to learn. This facilitates the more efficient use of advanced features of all bands rather than the use of several selected bands.

2) Superior precision and robustness. The results of our experiments demonstrate that the CNN-based detector works well when applied to different AVIRIS data, as it offers favorable nonlinear mapping and excellent learning capabilities in using convolutional neural networks.

3) Superior adaptability. The proposed detector not only detects targets of different sizes but also detects targets with differences in spectra. Compared to the eight classical detectors, the CNN-based target detector performs better.

More work must be conducted to reduce time consumption requirements and to render the detector applicable to data from different remote sensor.

REFERENCES

[1] Z. Li, J. Li, S. Zhou, and S. Pirasteh, "Comparison of spectral and spatial windows for local anomaly detection in hyperspectral imagery," *Int. J. Remote Sens.*, vol. 36, no. 6, pp. 1570–1583, 2015.
[2] L. Gao, B. Yang, Q. Du, and B. Zhang, "Adjusted spectral matched filter for target detection in hyperspectral imagery," *Remote Sens.*, vol. 7, no. 6, pp. 6611–6634, 2015.
[3] A. Averbuch and M. Zheludev, "Two linear unmixing algorithms to recognize targets using supervised classification and orthogonal rotation in airborne hyperspectral images," *Remote Sens.*, vol. 4, no. 2, pp. 532–560, Feb. 2012.
[4] L. Zhang, "Advance and future chanllenges in hyperspectral target detection," *Geomatics Inf. Sci. Wuhan Univ.*, vol. 39, no. 12, pp. 1387–1394, 2014.
[5] D. Manolakis and G. S. Shaw, "Detection algorithms for hyperspectral imaging applications," *IEEE Signal Process. Mag.*, vol. 19, no. 1, pp. 29–43, Jan. 2002.
[6] J. C. Harsanyi and C.-I. Chang, "Hyperspectral image classification and dimensionality reduction: An orthogonal subspace projection approach," *IEEE Trans. Geosci. Remote Sens.*, vol. 32, no. 4, pp. 779–785, Jul. 1994.
[7] C.-I. Chang, "Orthogonal subspace projection (OSP) revisited: A comprehensive study and analysis," *IEEE Trans. Geosci. Remote Sens.*, vol. 43, no. 3, pp. 502–518, Mar. 2005.
[8] C. Zhao, X. Jing, and W. Li, "Hyperspectral image target detection algorithm based on StOMP sparse representation," *J. Harbin Eng. Univ.*, vol. 36, no. 7, pp. 992–996, 2015.
[9] Y. Chen, N. M. Nasrabadi, and T. D. Tran, "Simultaneous joint sparsity model for target detection in hyperspectral imagery," *IEEE Geosci. Remote Sens. Lett.*, vol. 8, no. 4, pp. 676–680, Jul. 2011.
[10] A. K. Ziemann and D. W. Messinger, "An adaptive locally linear embedding manifold learning approach for hyperspectral target detection," *Proc. SPIE*, p. 94720O, May 2015.
[11] L. Zhang, L. Zhang, D. Tao, and X. Huang, "Sparse transfer manifold embedding for hyperspectral target detection," *IEEE Trans. Geosci. Remote Sens.*, vol. 52, no. 2, pp. 1030–1043, Feb. 2014.
[12] T. Wang, B. Du, and L. Zhang, "A background self-learning framework for unstructured target detectors," *IEEE Geosci. Remote Sens. Lett.*, vol. 10, no. 6, pp. 1577–1581, Nov. 2013.
[13] Y. Zhang, B. Du, and L. Zhang, "Regularization framework for target detection in hyperspectral imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 11, no. 1, pp. 313–317, Jan. 2014.
[14] Y. Liu, G. Gao, and Y. Gu, "Tensor matched subspace detector for hyperspectral target detection," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 4, pp. 1967–1974, Apr. 2017.
[15] Z. Li, S. Zhou, Y. Han, and L. Wang, "Local anomaly detection algorithm based on sliding windows in spectral space," *Proc. SPIE*, p. 92631G, Nov. 2014.
[16] H. Wu and S. Prasad, "Convolutional recurrent neural networks for hyperspectral data classification," *Sensors*, vol. 9, no. 3, p. 298, 2017.
[17] W. Li, G. Wu, F. Zhang, and Q. Du, "Hyperspectral image classification using deep pixel-pair features," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 2, pp. 844–853, Feb. 2017.
[18] L. Ran, Y. Zhang, W. Wei, and Q. Zhang, "A hyperspectral image classification framework with spatial pixel pair features," *Sensors*, vol. 17, no. 10, p. 2421, 2017.
[19] L. Zhang, Q. Zhang, B. Du, X. Huang, Y. Y. Tang, and D. Tao, "Simultaneous spectral-spatial feature selection and extraction for hyperspectral images," *IEEE Trans. Cybern.*, vol. 48, no. 1, pp. 16–28, Jan. 2018.
[20] W. Li, G. Wu, and Q. Du, "Transferred deep learning for anomaly detection in hyperspectral imagery," *IEEE Geosci. Remote Sens. Lett.*, vol. 14, no. 5, pp. 597–601, May 2017.
[21] R. O. Green *et al.*, "Imaging spectroscopy and the airborne visible/infrared imaging spectrometer (AVIRIS)," *Remote Sens. Environ.*, vol. 65, no. 3, pp. 227–248, Sep. 1998.
[22] Y. Zhang, K. Wu, B. Du, L. Zhang, and X. Hu, "Hyperspectral target detection via adaptive joint sparse representation and multi-task learning with locality information," *Remote Sens.*, vol. 9, no. 5, p. 482, 2017.

[23] C. E. Metz, "ROC methodology in radiologic imaging," *Investigative Radiol.*, vol. 21, no. 9, pp. 720–733, 1986.
[24] W. Li, Q. Du, and B. Zhang, "Combined sparse and collaborative representation for hyperspectral target detection," *Pattern Recognit.*, vol. 48, no. 12, pp. 3904–3916, 2015.
[25] M. Elad and M. Aharon, "Image denoising via sparse and redundant representations over learned dictionaries," *IEEE Trans. Image Process.*, vol. 15, no. 12, pp. 3736–3745, Dec. 2006.
[26] Q. Du, H. Ren, and C.-I. Chang, "A comparative study for orthogonal subspace projection and constrained energy minimization," *IEEE Trans. Geosci. Remote Sens.*, vol. 41, no. 6, pp. 1525–1529, Jun. 2003.
[27] X. Jin, S. Paswaters, and H. Cline, "A comparative study of target detection algorithms for hyperspectral imagery," *Proc. SPIE*, p. 73341W, Apr. 2009.
[28] T. F. Ayoub and A. R. Haimovich, "Modified GLRT signal detection algorithm," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 36, no. 3, pp. 810–818, Jul. 2000.
[29] J. Broadwater, R. Meth, and R. Chellappa, "A hybrid algorithm for subpixel detection in hyperspectral imagery," in *Proc. IEEE Int. Geosci. Remote Sens. Symp. (IGARSS)*, Sep. 2004, pp. 1601–1604.
[30] J. Broadwater and R. Chellappa, "Hybrid detectors for subpixel targets," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 11, pp. 1891–1903, Nov. 2007.

JINMING DU received the B.E. degree in engineering of surveying and mapping from the China University of Petroleum, Qingdao, China, in 2012. He is currently pursuing the M.S. degree with the Electronic Countermeasures Simulation and Evaluation Laboratory, National University of Defense Technology, Changsha, China. His research interests include remote sensing and deep learning in advanced technologies.

ZHIYONG LI received the B.E., M.S., and Ph.D. degrees in remote sensing from the National University of Defense Technology, Changsha, China, in 1997, 2000, and 2004, respectively. He spent one year as a Visiting Scholar with the University Of Waterloo, Waterloo, ON, Canada. He is currently an Associate Professor of photogrammetry and remote sensing with the National University of Defense Technology. He is also the Chief Technology Officer with Hunan Shenfan Technology Co., Ltd. His research interests include hyperspectral remote sensing and deep learning in advanced technologies.