Journal of Real-Time Image Processing
https://doi.org/10.1007/s11554-017-0742-z

SPECIAL ISSUE PAPER

# A real-time unsupervised background extraction-based target detection method for hyperspectral imagery

Cong Li[1,2] · Lianru Gao[1,3] · Yuanfeng Wu[1] · Bing Zhang[1,2] · Javier Plaza[4] · Antonio Plaza[4]

Received: 3 August 2017 / Accepted: 27 November 2017


**Abstract**
Target detection is an important technique in hyperspectral image analysis. The high dimensionality of hyperspectral data provides the possibility of deeply mining the information hiding in spectra, and many targets that cannot be visualized by inspection can be detected. But this also brings some problems such as unknown background interferences at the same time. In this way, extracting and taking advantage of the background information in the region of interest becomes a task of great significance. In this paper, we present an unsupervised background extraction-based target detection method, which is called UBETD for short. The proposed UBETD takes advantage of the method of endmember extraction in hyperspectral unmixing, another important technique that can extract representative material signatures from the images. These endmembers represent most of the image information, so they can be reasonably seen as the combination of targets and background signatures. Since the background information is known, algorithm like target-constrained interference-minimized filter could then be introduced to detect the targets while inhibiting the interferences. To meet the rapidly rising demand of real-time processing capabilities, the proposed algorithm is further simplified in computation and implemented on a FPGA board. Experiments with synthetic and real hyperspectral images have been conducted comparing with constrained energy minimization, adaptive coherence/cosine estimator and adaptive matched filter to evaluate the detection and computational performance of our proposed method. The results indicate that UBETD and its hardware implementation RT-UBETD can achieve better performance and are particularly prominent in inhibiting interferences in the background. On the other hand, the hardware implementation of RT-UBETD can complete the target detection processing in far less time than the data acquisition time of hyperspectral sensor like HyMap, which confirms strict real-time processing capability of the proposed system.

**Keywords** Hyperspectral imagery · Target detection · Unsupervised background extraction · Endmember extraction · Real-time processing · FPGA

✉ Lianru Gao
gaolr@radi.ac.cn

Cong Li
licong@radi.ac.cn

Yuanfeng Wu
wuyf@radi.ac.cn

Bing Zhang
zb@radi.ac.cn

Javier Plaza
jplaza@unex.es

Antonio Plaza
aplaza@unex.es

[1] Key Laboratory of Digital Earth Science, Institute of Remote Sensing and Digital Earth, Chinese Academy of Sciences, Beijing 100094, China

[2] University of Chinese Academy of Sciences, Beijing 100049, China

[3] The College of Computer Science and Software Engineering, Computer Vision Research Institute, Shenzhen University, Shenzhen 518060, China

[4] Hyperspectral Computing Laboratory, Department of Technology of Computers and Communications, Escuela Politecnica de Cáceres, University of Extremadura, Cáceres, Spain

Published online: 26 December 2017

## 1 Introduction

Target detection is one of the most important techniques in remote sensing analysis [1–4]. With very high spectral resolution, hyperspectral remote sensing provides significant information about the spectral characteristics of substances that cannot be visualized by inspection [1, 5]. For each pixel, it gives a nearly continuous spectrum and this reveals some underlying information, which brings better material recognition features. But due to the temporary reality that many onboard hyperspectral sensors' spatial resolution is not high enough while sometimes the target may be smaller, so the desired target in fact hides behind a so-called mixed pixel [6]. The spectrum of mixed pixel differs from the original after mixing, and this introduces additional noise effect on target detection. In many situations, some unwanted and unidentified signal sources may also be extracted [7]. Such unexpected results interfere with the accuracy of target detection and make the problem more complicated. Target detection using hyperspectral imagery has many applications in civilian and military aspects [8].

A simplified situation of target detection is anomaly detection, which identifies a small quantity of pixels that have significant differences from the most of rest pixels (which we called background) in spectral [9, 10]. We call this situation "unknown target, unknown background." But the target information sometimes is known, so we have to identify the particular item from the anomalies. There have been many algorithms developed for hyperspectral target detection. Adaptive coherence/cosine estimator (ACE) [11] and adaptive matched filter (AMF) [12] are two typical algorithms. ACE and AMF take advantage of the concept of generalized likelihood ratio test to estimate the probability of existence of targets. These algorithms work for the situation where the target of interest is known as a priori, while the background information is not. Another successful algorithm is constrained energy minimization (CEM) [13]. CEM derives from the linear constrained minimum variance beamformer in the field of digital signal processing, which picks up signal in particular direction and inhibits those in other directions. Using a unity constraint, CEM designs a finite impulse response (FIR) filter to pass the target while minimizing the average output power. CEM uses only target spectrum to solve the problem, but just suppressing the power of the whole image is not the best solution. If we had known some undesired targets or interferers, we should better eliminate the influence of these signals rather than minimizing the power. Another algorithm called target-constrained interference-minimized filter (TCIMF) [14] is designed to use both target and the known background. TCIMF calculates an operator than acts on both target and background, which can remove the interference of the undesired signatures prior to detection. TCIMF shows better performance than CEM under the scenario that both target and background are known, which is the most ideal situation for target detection problem. However, for most time, background is not known as a priori because of the fast changing of image scenes. So it is worth to develop a method that can extract the background information automatically from the image without human intervention. Such method will turn the situation from "known target, unknown background" to "known target, known background" and takes full advantage of the wealth of information that hyperspectral imagery provides.

Hyperspectral remote sensing shows the above-mentioned advantages in deep-level information extraction, but at the same time brings about some computational problems. To get more detailed information, this technology is making its way to more improved spectral, spatial and temporal resolution, which means faster computing solutions is demanded as well [15–17]. For some applications, real-time image processing is required to achieve quick response to the area of interest. On the other hand, many hyperspectral sensors are set on satellites, where the bandwidth of the downlink channel is limited. Just downloading the target detection result instead of the huge amount of original data is a more effective way. Onboard real-time image processing with hardware accelerators helps to achieve such goal and may play an important role in building the intelligent remote sensing satellite system in the future [18]. Commodity graphics processing units (GPUs) [19] and field programmable gate arrays (FPGAs) [20–22] are two typical specialized hardware devices, and both provide highly desirable features such as low weight and high computing performance, but FPGAs can also offer reconfigurability and lower energy consumption, in addition to radiation tolerance that still cannot be achieved by GPUs [23]. An FPGA is an integrated circuit whose logic blocks can be defined and configured by users repeatedly. That means that FPGAs offer the possibility of adaptively selecting a proper algorithm or new set of parameters to be applied from the control station on Earth anytime [24]. FPGAs also exhibit an architecture that is suitable to implement pipeline processing [25]. Some fantastic recent works have explored FPGA-based implementations of target detection algorithms. For instance, the aforementioned algorithm CEM has been implemented on FPGA in [26], along with some other target detection algorithms in [27–29].

In this paper, we present an unsupervised background extraction-based target detection method, which is called UBETD for short. The concept of hyperspectral unmixing is used to extract the background from the hyperspectral

image. A simplified version suitable for hardware implementation is also presented as RT-UBETD. There are three important advantages for this algorithm: (1) it uses hyperspectral unmixing to automatically extract background information, turning the scenario to "known target, known background" situation and then removing the interference of these undesired signatures; (2) its computational complexity is reduced and amenable for strictly real-time processing; (3) its algorithm structure is suitable for being implemented on FPGAs. The newly developed RT-UBETD has been implemented on a Virtex-7 FPGA evaluation board and tested using both synthetic and real hyperspectral images. The obtained results show that the proposed algorithm exhibits better performance than CEM. The main contributions of this paper are as follows:

1. Endmember extraction and target detection are both important techniques for hyperspectral image analysis. We propose a new strategy applying endmember extraction in target detection applications to reveal the underlying background information. The proposed strategy can turn a "known target, unknown background" problem to an easier solving "known target, known background" problem.
2. The new unsupervised background extraction method is universally applicable for any endmember extraction algorithm in theory. This paper shows an example algorithm for illustration, in which RT-MSVA is used for background extraction while TCIMF is used for the following target detection. Such algorithm is named as UBETD.
3. The proposed algorithm is implemented on a Virtex-7 FPGA. A new block-iterative inverse algorithm is introduced to further simplify the calculation of inverse matrixes in UBETD. The PFGA-based implementation proves that it can achieve strict real-time capability. The new block-iterative inverse algorithm also is valuable to other real-time target detection methods which have the same requirement of calculating inverse matrixes.

The remainder of this paper is organized as follows. Section 2 describes some related algorithms, such as TCIMF and real-time maximum simplex volume algorithm (RT-MSVA). Section 3 describes the UBETD algorithm, including background extraction by using endmember extraction method. Section 4 describes simplified RT-UBETD and its hardware implementation, providing details about the architecture of the complete system and the main processing modules. In Sect. 5, we firstly describe the synthetic and real images used in the experiments and then show the processing results and analyses. The computational performance is also evaluated in this section. Finally, Sect. 6 concludes the paper and discusses some assumptions for future research.

## 2 Related algorithms

In this section, we provide a brief overview of some related algorithms in this paper. TCIMF is a widely used target detection algorithms using target signatures and background signatures together. RT-MSVA is an endmember extraction algorithm using geometrical method which takes full advantage of band information without dimensionality reduction.

### 2.1 TCIMF

TCIMF can be seen as an extension of CEM [14]. Firstly we give a brief description of CEM. The CEM algorithm is suitable when the spectrum for target of interest is known but the background is unknown [13]. This algorithm is based on the method of linear constrained minimum variance beamformer in the field of digital signal processing, which extracts signals in particular direction and inhibits those in other directions. The hyperspectral image can be described as $\mathbf{X} = \{\mathbf{x}_i\}_{i=1}^{n}$ and the target vector is $\boldsymbol{d}$. CEM designs a finite impulse response filter vector $\mathbf{w} = (\mathbf{w}_1, \mathbf{w}_2, \ldots, \mathbf{w}_L)^{\mathrm{T}}$ that minimizes the output energy under the constraint of $\mathbf{w}^{\mathrm{T}}\mathbf{d} = 1$. The variable $L$ stands for the number of bands in the hyperspectral image, and $n$ is the total number of pixels. Let input be $\mathbf{x}_i$, detection statistics be $y_i$, then $y_i = \mathbf{w}^{\mathrm{T}}\mathbf{x}_i = \mathbf{x}_i^{\mathrm{T}}\mathbf{w}$. In this way, the average output energy would be

$$\frac{1}{n}\left[\sum_{i=1}^{n} y_i^2\right] = \frac{1}{n}\left[\sum (\mathbf{x}_i^{\mathrm{T}}\mathbf{w})^{\mathrm{T}}\mathbf{x}_i^{\mathrm{T}}\mathbf{w}\right] = \mathbf{w}^{\mathrm{T}}\left(\frac{1}{n}\left[\sum \mathbf{x}_i\mathbf{x}_i^{\mathrm{T}}\right]\right)\mathbf{w} = \mathbf{w}^{\mathrm{T}}\mathbf{R}\mathbf{w} \tag{1}$$

where $\mathbf{R}$ is the self-correlation matrix of original image data. Obviously, the self-correlation matrix $\mathbf{R}$ will turn to covariance matrix $\sum$ if removing the mean vector from $\mathbf{x}$. A minimum problem is introduced to get the filter vector $\mathbf{w}$.

$$\begin{cases} \min\limits_{\mathbf{w}}\left(\frac{1}{n}\left[\sum\limits_{i=1}^{n} y_i^2\right]\right) = \min\limits_{\mathbf{w}}(\mathbf{w}^{\mathrm{T}}\mathbf{R}\mathbf{w}) \\ \mathbf{w}^{\mathrm{T}}\mathbf{d} = 1 \end{cases} \tag{2}$$

Solve this extreme problem by Langrage operator, and the solution is CEM detector [13].

$$\mathbf{w}^{\mathrm{CEM}} = \frac{\mathbf{R}^{-1}\mathbf{d}}{\mathbf{d}^{\mathrm{T}}\mathbf{R}^{-1}\mathbf{d}} \tag{3}$$

Background interference plays an important role in hyperspectral target detection. Although more subtle materials can be revealed by high spectral resolution, there are also some undesired signal sources extracted at the same time. CEM minimizes the average output energy to suppress the background, but in some situations in which the background interferers are available, ignoring these useful information is not effective. Still, in some applications there are more than one target desired, and we hope to extract these targets simultaneously. TCIMF is designed to solve the aforementioned problems.

In TCIMF algorithm, the known signatures are divided into two classes. One is $\mathbf{D} = [\mathbf{d}_1, \mathbf{d}_2, \ldots, \mathbf{d}_{n_D}]$ that denotes the target signature matrix, and the other is $\mathbf{U} = [\mathbf{u}_1, \mathbf{u}_2, \ldots, \mathbf{u}_{n_U}]$ that denotes the undesired material signature matrix, where $n_D$ and $n_U$ are the number of the target signatures and the number of the undesired material signatures, respectively. Then, the detector $\mathbf{w}^{\mathrm{TCIMF}}$ is designed to deal with $\mathbf{D}$ and $\mathbf{U}$ at the same time using the following equation:

$$[\mathbf{D}, \mathbf{U}]^{\mathrm{T}}\mathbf{w} = \begin{bmatrix} \mathbf{1}_{n_D \times 1} \\ \mathbf{0}_{n_U \times 1} \end{bmatrix} \tag{4}$$

where $\mathbf{1}_{n_D \times 1}$ is an $n_D \times 1$ unity vector to filter the target signature matrix $\mathbf{D}$, and $\mathbf{0}_{n_U \times 1}$ is an $n_U \times 1$ zero vector to annihilate the undesired material signature matrix $\mathbf{U}$. Then, the minimum problem in Eq. (2) can be extended to the following linear constrained optimization problem:

$$\begin{cases} \min\limits_{\mathbf{w}} (\mathbf{w}^{\mathrm{T}}\mathbf{R}\mathbf{w}) \\ [\mathbf{D}, \mathbf{U}]^{\mathrm{T}}\mathbf{w} = \begin{bmatrix} \mathbf{1}_{n_D \times 1} \\ \mathbf{0}_{n_U \times 1} \end{bmatrix} \end{cases} \tag{5}$$

Then, we use the same solution to solve the problem and get the TCIMF detector. $\mathbf{w}^{\mathrm{TCIMF}}$ is given by

$$\mathbf{w}^{\mathrm{TCIMF}} = \mathbf{R}^{-1}[\mathbf{D}, \mathbf{U}]\Big([\mathbf{D}, \mathbf{U}]^{\mathrm{T}}\mathbf{R}^{-1}[\mathbf{D}, \mathbf{U}]\Big)^{-1}\begin{bmatrix} \mathbf{1}_{n_D \times 1} \\ \mathbf{0}_{n_U \times 1} \end{bmatrix} \tag{6}$$

where $\mathbf{R}$ is the self-correlation matrix. Applying this detector vector can get TCIMF target detection result

$$y = \mathbf{D}_{\mathrm{TCIMF}}(\mathbf{x}) = \left(\mathbf{w}^{\mathrm{TCIMF}}\right)^{\mathrm{T}}\mathbf{x}. \tag{7}$$

TCIMF uses background information to inhibit the interference and is a good choice for target detection after background extraction.

## 2.2 RT-MSVA

RT-MSVA is an endmember extraction algorithm developed based on geometrical methods [30]. Under the assumption of linear mixture model (LMM) [31] which ignores multiple scattering effects between endmembers, the hyperspectral image can be described as follows:

$$\mathbf{X} = \mathbf{E} \times \mathbf{A} + \mathbf{W} \tag{8}$$

where $\mathbf{X} = [x_1, x_2, \ldots, x_n]$ is a hyperspectral image with a total of $n$ pixels, $\mathbf{E} = [\mathbf{e}_1, \mathbf{e}_2, \ldots, \mathbf{e}_p]$ is the endmember matrix, $\mathbf{A} = [\mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_n]$ is the abundance matrix and $\mathbf{W} = [\mathbf{w}_1, \mathbf{w}_2, \ldots, \mathbf{w}_n]$ is an error matrix.

Each pixel in the hyperspectral image corresponds to a point in the high-dimensionality space. Geometrical methods are based on the assumption that the mixed pixels lie inside of a simplex in which the corners are the endmembers. In other words, the endmembers are assumed to be on the vertices of the data simplex. RT-MSVA uses a simplex growing method to find a set of pixels with the maximum simplex volume. Firstly, in the initialization step, the pixel with the maximum length is extracted as the first endmember. Then, the algorithm traverses all the pixels to find the one that provides the largest simplex volume together with the former extracted endmembers, and this resulting pixel is then extracted as the new endmember. This process will be done repeatedly until all the $p$ endmembers are already got. The main difficulty in calculation is the simplex volume. Different from some typical geometrical model-based algorithms like N-FINDR [32] and SGA [33], RT-MSVA uses a new simplex volume formula taking full advantage of all the bands without dimensionality reduction. The volume of simplex formed by endmembers $\{\mathbf{e}_1, \mathbf{e}_2, \ldots, \mathbf{e}_p\}$ is calculated as follows:

$$\mathbf{A} = \mathbf{W}^{\mathbf{T}}\mathbf{W} \tag{9}$$

$$V(\mathbf{e}_1, \mathbf{e}_2, \ldots, \mathbf{e}_p) = \frac{1}{(p-1)!}\sqrt{|\det(\mathbf{A})|} \tag{10}$$

where $\mathbf{W} = [\mathbf{e}_2 - \mathbf{e}_1, \mathbf{e}_3 - \mathbf{e}_1, \ldots, \mathbf{e}_p - \mathbf{e}_1]$, $\mathbf{A}$ is the measure matrix. Woodbury's identity is introduced in matrix determinant calculation in (10) to simplify the computational complexity.

RT-MSVA needs no dimensionality reduction process and has admirable performance in endmember extraction. Its characteristic is quite suitable for applying in target detection which we will discuss later in Sect. 3. Also, it is amenable for strictly real-time processing. It can be summarized by the algorithmic description given in Algorithm 1

**Algorithm 1:** RT-MSVA

**Input**: hyperspectral image data $\mathbf{r}$, number of endmembers $p$.

Step 1: initialization: calculate the vector length of each pixel r, the pixel that provides the maximum length is the first endmember $\mathbf{e}_1$. Let $i = 0$.

Step2: let $i = i + 1$, and start the inner loop.

Step3: For each pixel $\mathbf{r}$, calculate the volume $V(\mathbf{e}_1, ..., \mathbf{e}_{i-1}, \mathbf{r})$ defined by (9) and (10).

Step4: A variable det_max is used to store the largest determinant within this inner loop. Compare $V(\mathbf{e}_1, ..., \mathbf{e}_{i-1}, \mathbf{r})$ with det_max, if $V(\mathbf{e}_1, ..., \mathbf{e}_{i-1}, \mathbf{r}) > det\_max$, let $det\_max = V(\mathbf{e}_1, ..., \mathbf{e}_{i-1}, \mathbf{r})$. The pixel that provides the largest $V(\mathbf{e}_1, ..., \mathbf{e}_{i-1}, \mathbf{r})$ is extracted as endmember $\mathbf{e}_i$.

Step5: if $i \leq p$, then go to step 2. Otherwise, the subset $\{\mathbf{e}_1, \mathbf{e}_2, \ldots, \mathbf{e}_p\}$ is the set of desired p endmembers.

**Output:** endmembers $\{\mathbf{e}_1, \mathbf{e}_2, \ldots, \mathbf{e}_p\}$.

.

## 3 UBETD algorithm

### 3.1 Background extraction using endmember extraction method

The basic theory of target detection is highlighting the target while suppressing the background. In the situation where the target spectrum is known as a priori, estimating the background information will be the first step in the process. In the traditional algorithms, background information is always estimated as a whole via approaches like calculating the covariance matrix of the hyperspectral image. This method stands on the assumption of the sparsity of the interesting target in the hyperspectral image, which means that the number of target pixels is extremely small compared to the total pixel number. But even so, this undirected way still includes the target information that will cause errors in the results. On the other hand, there are interferers together with pixels of less relationship with the target in the remaining pixels, and treating them in a same way cannot effectively remove the influence of interferers. We present a method to extract the specific spectra of background so that the following process can deal with the real background information excluding the target and then remove the influence of these undesired signatures.

Endmember extraction in spectral unmixing is a technique that picks up pure signature pixels from the hyperspectral image which are called endmembers. A properly extracted set of endmembers can represent the whole information of the image scene, which can be reasonably seen as the combination of targets and background material signatures. The number of endmembers to extract can be calculated by an eigen decomposition-based algorithm HySime [34], which firstly estimates the signal and noise correlation matrices and then selects the subset of eigenvalues that best represents the signal subspace in the least squared error sense. It should be noted that the HySime algorithm has a FPGA-based real-time implementation in [35]. Such background extraction method using endmember extraction has the following advantages:

1. Comprehensive coverage of information. Endmembers are material signatures that represent most of the hyperspectral image information. They can cover every material within the scene in theoretical situation. So the background we extracted contains most of the information besides the desired target which may result in better performance in target detection.
2. Diversity of background signatures extracted. Endmembers usually have big differences from each other. Such as geometrical-based endmember extraction algorithms, the endmembers extracted are vertices of the simplex with the maximum volume, which means that they are quite far from each other in high-dimensionality space. This guarantees the diversity of the background signatures by avoiding repeat information extraction.
3. Automation. The method is completely unsupervised that all the operations are implemented automatically without any kind of human intervention. The proposed method can apply to any hyperspectral image in any situation to detect the specific targets desired. This is also a brilliant characteristic for onboard systems.

According to the characteristics of the background extraction process we design, many endmember extraction algorithms can be used for such purpose. A state-of-the-art algorithm RT-MSVA is chosen as an example to apply such process in this paper. Besides the advantages mentioned above, RT-MSVA also has some good characteristics itself, in which the most attractive is that it can be applied with all the bands of the original hyperspectral image without dimensionality reduction. Avoiding the loss of information while reducing dimensionality is not only beneficial for endmember extraction, but can also lead to a better performance in the following target detection. On the other hand, RT-MSVA is suitable for real-time processing. A FPGA implementation of RT-MSVA has been presented by our former work in [30]. RT-MSVA follows the simplex growing method and extracts the endmembers one by one. If we let the targets $\{\mathbf{d}_1, \mathbf{d}_2, \ldots, \mathbf{d}_{n_D}\}$ be the first $n_D$ endmembers in the initialization step of RT-MSVA, then the following extracted endmembers which represent the image together with the targets can be seen as the

background signatures. In this way, we can get the specific background signatures via an endmember extraction process without human intervention.

### 3.2 UBETD

As mentioned above, RT-MSVA and TCIMF are chosen as an illustration of the proposed UBETD in this paper. To decide how many background signatures to be extracted is important for the whole processing chain, because if excess signatures extracted as background may obtain repeated information. Especially when some signatures similar to the target are misplaced in the undesired material signature matrix $\mathbf{U}$, it may cause extra error to the detection result. On the other hand, too few background signatures extracted make the background information incomplete, which is also not effective. Such estimation of number of endmembers can be done by the aforementioned algorithm HySime, which also has an FPGA implementation version. Based on this outstanding work, the number of endmembers can be achieved automatically, so we regard this number $n_E$ as a known parameter for our target detection algorithm. As for the number of targets $n_D$ is also known as a priori, so the number of background material signatures can be easily calculated as $n_U = n_D - n_E$.

RT-MSVA is introduced into this process as an endmember extraction algorithm to achieve the background information. Setting the target signatures $\{\mathbf{d}_1, \mathbf{d}_2, \ldots, \mathbf{d}_{n_D}\}$ as the first $n_D$ endmembers, we then get the desired endmembers as $\{\mathbf{d}_1, \mathbf{d}_2, \ldots, \mathbf{d}_{n_D}, \mathbf{e}_{n_D+1}, \ldots, \mathbf{e}_{n_E}\}$. This set of endmembers will then be divided into two classes for TCIMF, in which the target signature matrix is $\mathbf{D} = [\mathbf{d}_1, \mathbf{d}_2, \ldots, \mathbf{d}_{n_D}]$, and the undesired material signature matrix is $\mathbf{U} = [\mathbf{e}_{n_D+1}, \ldots, \mathbf{e}_{n_E}]$. It is easy to find that the sample matrix $[\mathbf{D}, \mathbf{U}]$ for TCIMF is just the matrix formed by the former extracted endmember set, so we let $\mathbf{E} = [\mathbf{d}_1, \mathbf{d}_2, \ldots, \mathbf{d}_{n_D}, \mathbf{e}_{n_D+1}, \ldots, \mathbf{e}_{n_E}]$ be the sample matrix for UBETD. Then, we use the following equation to get the UBETD detector:

$$\mathbf{w}^{\mathrm{UBETD}} = \mathbf{R}^{-1}\mathbf{E}\left(\mathbf{E}^{\mathrm{T}}\mathbf{R}^{-1}\mathbf{E}\right)^{-1}\begin{bmatrix}\mathbf{1}_{n_D\times 1}\\ \mathbf{0}_{n_U\times 1}\end{bmatrix} \tag{11}$$

where $\mathbf{R}$ is the self-correlation matrix, $\mathbf{1}_{n_D\times 1}$ is a $n_D \times 1$ unity vector and $\mathbf{0}_{n_U\times 1}$ is a $n_U \times 1$ zero vector. At last, the UBETD detector $\mathbf{w}^{\mathrm{UBETD}}$ is applied to all the pixel vectors in original image data $\mathbf{X}$ to get the final target detection result

$$\mathbf{Y} = D_{\mathrm{UBETD}}(\mathbf{X}) = \left(\mathbf{w}^{\mathrm{UBETD}}\right)^{\mathrm{T}}\mathbf{X} \tag{12}$$

The proposed UBETD algorithm can realize unsupervised background information extraction, which turns a "known target, unknown background" problem to an easier solving "known target, known background" problem. The algorithm can be summarized by the algorithmic description given in Algorithm 2.

**Algorithm 2:** UBETD

**Input:** hyperspectral image data $\mathbf{X}$, number of endmembers $n_E$, target signatures $\{\mathbf{d}_1, \mathbf{d}_2, \ldots, \mathbf{d}_{n_D}\}$.

Step1: Initialization: set the target signatures $\{\mathbf{d}_1, \mathbf{d}_2, \ldots, \mathbf{d}_{n_D}\}$ as the first $n_D$ endmembers, and the number of background signatures is $n_U = n_D - n_E$.

Step2: Run the RT-MSVA to extract all the other $n_U$ endmembers. The resulting set of endmembers is $\{\mathbf{d}_1, \mathbf{d}_2, \ldots, \mathbf{d}_{n_D}, \mathbf{e}_{n_D+1}, \ldots, \mathbf{e}_{n_E}\}$.

Step3: Let $\mathbf{E} = [\mathbf{d}_1, \mathbf{d}_2, \ldots, \mathbf{d}_{n_D}, \mathbf{e}_{n_D+1}, \ldots, \mathbf{e}_{n_E}]$ to be the sample matrix, which is the matrix formed by the endmember set extracted in step2, and then calculate the UBETD detector $\mathbf{w}^{\mathrm{UBETD}}$ defined by (11).

Step4: Apply the detector to $\mathbf{X}$ using equation (12) and get the detection result $\mathbf{Y}$.

**Output:** Target detection result $\mathbf{Y}$.

## 4 RT-UBETD and its FPGA implementation

### 4.1 RT-UBETD

The UBETD detector calculation procedure represents most of the computation in the proposed target detection algorithm. From Eq. (11) in Sect. 3, we can see that two inverse matrices will be calculated, one is a $L \times L$ matrix and the other is a $n_E \times n_E$ matrix. Calculating the inverse matrix involves significant computational complexity, especially for the self-correlation matrix with the size of $L \times L$ for the reason that band number $L$ is usually very large for hyperspectral images. Some approaches like QR decomposition and LU decomposition have been used for FPGA to solve this problem [36, 37]. To further simplify the calculation, we introduce a new block-iterative inverse algorithm for large-scale real symmetric matrixes [38].

For a $t \times t$ real symmetric matrix $\mathbf{W}_t$, it can be partitioned as follows:

$$\mathbf{W}_t = \begin{bmatrix}\mathbf{W}_{t-1} & \mathbf{r}_t\\ \mathbf{r}_t^{\mathrm{T}} & p_t\end{bmatrix} \tag{13}$$

where $\mathbf{W}_{t-1}$ is the $(t-1)$-order square matrix, $\mathbf{r}_t$ is a $(t-1) \times 1$ vector and $p_t$ is the last element in the matrix. If the inverse of $\mathbf{W}_{t-1}$ is known, then the inverse of $\mathbf{W}_t$ can be calculated by the following formulas:

$$\begin{cases} \mathbf{b}_t = -\mathbf{W}_{t-1}^{-1}\mathbf{r}_t \\ \beta_t = p_t + \mathbf{r}_t^{\mathrm{T}}\mathbf{b}_t \\ \alpha_t = \dfrac{1}{\beta_t} \\ \mathbf{q}_t = \alpha_t \mathbf{b}_t \end{cases} \tag{14}$$

$$\mathbf{W}_t^{-1} = \begin{bmatrix} \mathbf{W}_{t-1}^{-1} + \mathbf{q}_t\mathbf{b}_t^{\mathrm{T}} & \mathbf{q}_t \\ \mathbf{q}_t^{\mathrm{T}} & \alpha_t \end{bmatrix} \tag{15}$$

where $\mathbf{b}_t$ and $\mathbf{q}_t$ are $(t-1)\times 1$ vectors, $\beta_t$ and $\alpha_t$ are variables. In this way, the inverse of a $t \times t$ matrix can be calculated easily by the inverse of its $(t-1)\times(t-1)$ submatrix. We can regard the variable $t$ as the loop variable, and for $t = 1$, the matrix $\mathbf{W}_t$ only has 1 element and the inverse $\mathbf{W}_t^{-1}$ is its reciprocal. Then, we operate this procedure repeatedly while adding 1 to the order of matrix each time, until $t = L$ and the inverse of a $L \times L$ matrix has been achieved.

This algorithm is quite faster than QR decomposition and LU decomposition when applied on large-scale matrixes, and more suitable for FPGA implementation. We use RT-UBETD to represent the simplified algorithm. This RT-UBETD is implemented on an FPGA board as described below.

### 4.2 FPGA implementation of RT-UBETD

Figure 1 shows the hardware architecture of the RT-UBETD processing system. A DDR3 SDRAM (synchronous dynamic random access memory) outside the FPGA chip is used to store the whole hyperspectral image. The target signature matrix $\mathbf{D}$ is also stored in it at a specific location. A memory interface generator (MIG) is used to control the SDRAM and register the input data with the help of a first-in-first-out (FIFO) queue. Reading data from the SDRAM costs several clock periods but sometimes processing such data can be faster than that. To avoid letting the system be idle waiting for data and ensure a complete processing workflow, the FIFO has been set to contain two complete rows of the hyperspectral image which builds a ping-pong buffer. Every time a row has been processed (i.e., the FIFO is half empty), a new data row is brought in. Each pixel in the hyperspectral images will be calculated $p$ times in the background extraction stage and 2 times in the target detection stage, so the FIFO needs to read the image data $p + 2$ times throughout the whole process.

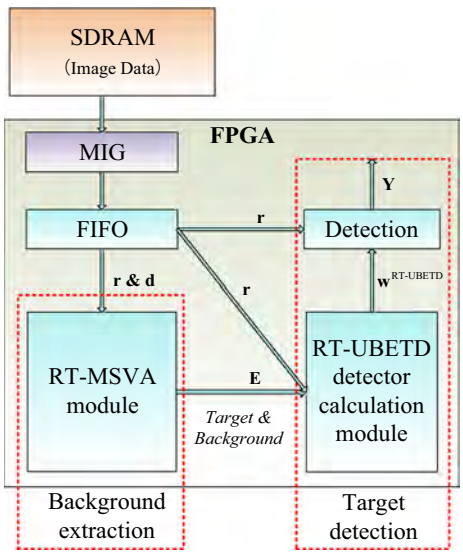



**Fig. 1** Hardware architecture for the FPGA implementation of RT-UBETD

The whole processing chain can be summarized to two stages, the first is shown in the left part of Fig. 1 which is the background extraction stage, and the second is the target detection stage shown in the right part. The RT-MSVA module performs endmember extraction to get the background material signatures. This module has been detailed in our former work [30]. The RT-UBETD detector calculation module plays an important role that achieves the RT-UBETD detector from image data and the sample matrix $\mathbf{E}$. The RT-MSVA module needs to traverse all the pixels in the image for $n_U$ times to get the sample matrix $\mathbf{E}$. At the same time, the FIFO also sends image pixels to the RT-UBETD detector calculation module to calculate the self-correlation matrix $\mathbf{R}$ and its inverse matrix $\mathbf{R}^{-1}$ in advance. Because of the hardware characteristic of FPGA, these two different modules are constructed in different areas using distinct resources, respectively, so they can work at the same time without mutual interference. After stage one, the sample matrix is sent to the next module to start stage two. In the last, the detector $\mathbf{w}^{\mathrm{RT-UBETD}}$ is applied to the image data in detection module and results the detection matrix $Y$. The final output is a gray scale image defined by $Y$ which is temporarily stored in a block RAM.

The data flow of the block-iterative inverse calculation procedure is shown in Fig. 2. The abbreviations in the figure response to adder, subtracter, multiplier, and divider, respectively. We use the IP Core Generator to generate some of these components like multipliers and dividers. We can choose between using look-up tables (LUTs) or digital signal processors (DSPs) to construct these components in a graphical interface.

Floating point format is normally used in CPU-based computer systems and may have some advantages in accuracy for mathematical operations, but it costs too many resources for FPGAs. In fact, floating point numbers are also transformed to binary integers in processing. So we used fixed-point format in our implementation. Figure 3 illustrates the adopted data structure in fixed-point format.

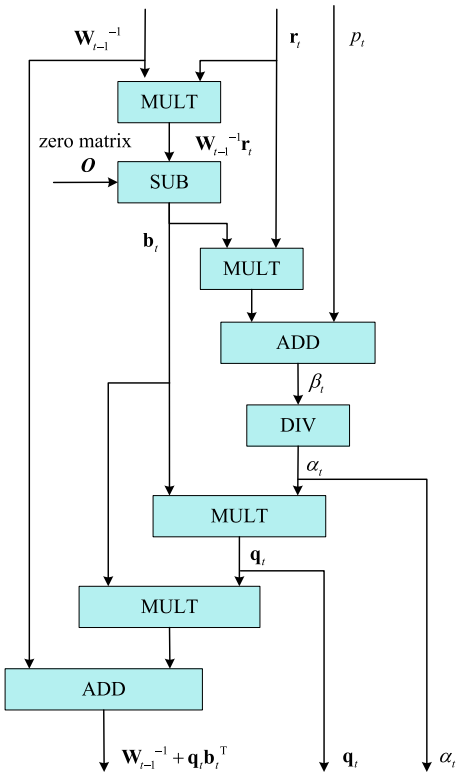



**Fig. 2** Data flow of the block-iterative inverse calculation procedure

The input data are given by reflectance units which are usually smaller than 1, so we use a 16-bit binary format to represent the data. The highest bit is the sign, and the next bit expresses the integer part while the lower 14 bits express the fractional part. To store the elements in inverse matrix is a more difficult problem because of the wide range in data value, so we need to find a compromise for the size of the data. Especially for inverse matrix calculation procedure in which the iteration method is used, the error may be bigger and bigger with the increase in the loop index. As a result, a 48-bit format with 32 bits as fractional part is used to represent elements in the inverse matrix. Enough fractional bits can control the error caused by cutting off some bits while computing.

The hardware architecture has been implemented on a Xilinx VC709 connectivity kit with a Virtex-7 XC7VX690T-2FFG1761C FPGA (see Fig. 4). The Virtex-7 family of FPGAs provides the highest performance and

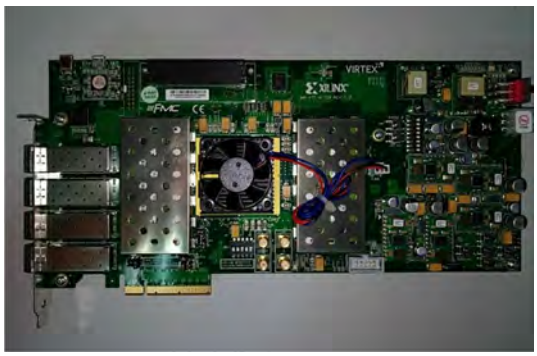


**Fig. 4** Xilinx VC709 connectivity kit with a XC7VX690T-2FFG1761C FPGA

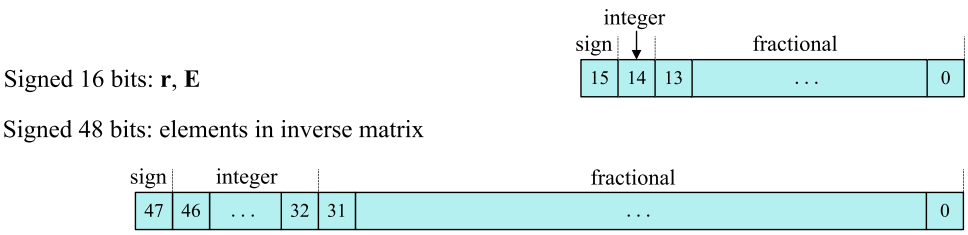
integration with 28-nm technology. The XC7VX690T FPGA is a high-end product in Virtex-7 series which has 108,300 slices, 693,120 logic cells and 866,400 CLB flip-flops available, and also a total block RAM of 52,920 kb. It also offers 3600 DSP slices which is the most among the mainstream products of Xilinx. Each DSP slices contains a pre-adder, a 25 × 18 multiplier, an adder and an accumulator. This is an outstanding advantage for large-scale data computing.

## 5 Experimental results and analysis

In this section, we will discuss the performance of the proposed UBETD and the FPGA implementation of RT-UBETD. Experiments with CEM, ACE and AMF are also performed for comparison. To directly compare these algorithms, we perform only one target of interest in the comparative experiments. A multi-target detection using UBETD is also performed in the last to illustrate its ability of detecting several targets simultaneously.

The hardware experiment is performed on the aforementioned VC709 connectivity kit. The architecture is implemented using the VHDL language. We use ISE 14.2 as the developing environment, and Modelsim SE 10.1a to perform simulation. The comparative experiments are conducted on MATLAB in an Intel Core i3 CPU, with a working frequency of 3.5 GHz. Both synthetic and real hyperspectral images will be used to assess target detection accuracy. The HyMap data are also used to analyze the

**Fig. 3** Data structure used in the FPGA implementation



computational performance and the resource utilization of the hardware implementation.

The remainder of this section is organized as follows. In subsection A, we describe the synthetic and real hyperspectral images that will be used in the experiments. In subsection B, we introduce the metrics used to assess the target detection performance. In subsection C, we discuss the obtained results and provide an evaluation of target detection accuracy in comparison with the CEM, ACE and AMF. The result of multi-target detection is also shown in this subsection. Subsection D provides an analysis of the computational performance of our hardware implementation.

## 5.1 Hyperspectral image data sets

### 5.1.1 Synthetic hyperspectral image

We designed a synthetic image to clearly demonstrate the function of our algorithm. Five spectral signatures of different materials are chosen from the USGS spectral library: fir tree, blackbrush, blue spruce, dry grass and annite. All these spectral signatures have been resampled to 189 bands with wavelength range from 0.4 to 2.5 μm, removing the bands with low signal-to-noise ratio. The spectral signature of fir tree and blackbrush is much similar to each other, so we take blackbrush as the target desired and the fir tree as the main interferer. Blue spruce is another slighter interferer, while dry grass and annite are background with totally different spectral signatures. To better show the performance of our proposed algorithm, we have made the synthetic image scene a relatively extreme case: (1) the targets have very low rate of existence; (2) there are pure pixels of the four background materials and the desired target; and (3) the background is homogeneous with ubiquitous interference in the whole image. The location of targets and interferers makes no difference to the results because the proposed algorithm uses the spectral statistic information but not spatial information. In this way, an image of $50 \times 50$ pixels is made by the following approaches: the homogeneous background pixels consist of 10% of annite, 10% of dry grass, 20% of blue spruce and 60% of fir tree; then 9 targets are inserted in (10, 5), (10, 10),…, (10, 45) with abundance of 5, 10, 20, 30, 40, 50, 60, 80 and 100%, respectively; at last, 6 interferers are inserted in (30, 5), (30, 10),…, (30, 30), in which the former 4 interferers are pure pixels of background materials and the last 2 contain 90 and 80% of fir tree, respectively. The abundance of original background in these pixels is scaled down according to the abundance of targets or interferers inserted. An additive white Gaussian noise (AWGN) is added into this synthetic image, and the SNR is 15 dB. Figure 5a shows the 3D data cube of the synthetic image, and Fig. 5b–d shows the spectra of background, interferers and targets, respectively. Figure 5e is the ground truth of the synthetic image.

### 5.1.2 Real hyperspectral images

The first real hyperspectral image in this experiment is a HyMap reflectance image of Cook City in Montana, USA. The data set was provided as part of a publicly available self-test exercise for target detection purposes by the Digital Imaging and Remote Sensing Group, Center of Imaging Science, Rochester Institute of Technology. This data set has $800 \times 280$ pixels and covers the target region, the mountain region and part of the more complex urban region (see Fig. 6a). This data set comprises 126 bands from 0.4 to 2.4 μm, and the spatial solution is about 3 m. A target of red cotton was placed in the grass area near the center of the image. The data set provides the SPL file and the exact location of the target, which makes it a standard test site for evaluating the accuracy of hyperspectral target detection algorithms. Figure 6b, c shows the ground truth and the target signature, respectively.

The second real data set was taken over the Rochester Institute of Technology by the ProSpecTIR-VS2 hyperspectral sensor. This sensor consists of two important parts, an imager that covers the visible and near infrared spectra from 0.39 to 1.0 μm, and a short-wave infrared imager covers 1.0–2.5 μm. The data set was achieved by the SpecTIR Hyperspectral Airborne Rochester Experiment (SHARE) program. We take a subset over a parking lot with the size of $180 \times 180$ pixels and 126 bands (see Fig. 7a). Two kinds of artificial targets with three different sizes were placed in such area, which are red cotton and blue cotton. Figure 7b shows the ground truth in which red cotton is shown in red and blue cotton is shown in blue.

## 5.2 Assessment metrics

1. *ROC and AUC* the receiver operating characteristic (ROC) curves establish a one-to-one correspondence between the true positive ratio (TPR) and the false positive ratio (FPR) [39]. TPR usually refers to the probability of correctly rejecting the null hypothesis for a particular test. In hyperspectral target detection, the true positive rate is calculated as the ratio between the number of correctly detected target pixels and the actual number of target pixels. FPR refers to the probability of falsely rejecting the null hypothesis and is calculated as the ratio between the number of falsely detected target pixels and the actual number of background pixels. This metric offers quantitative evaluation of the performance of target detection algorithms, by the way that the algorithm corresponding to the

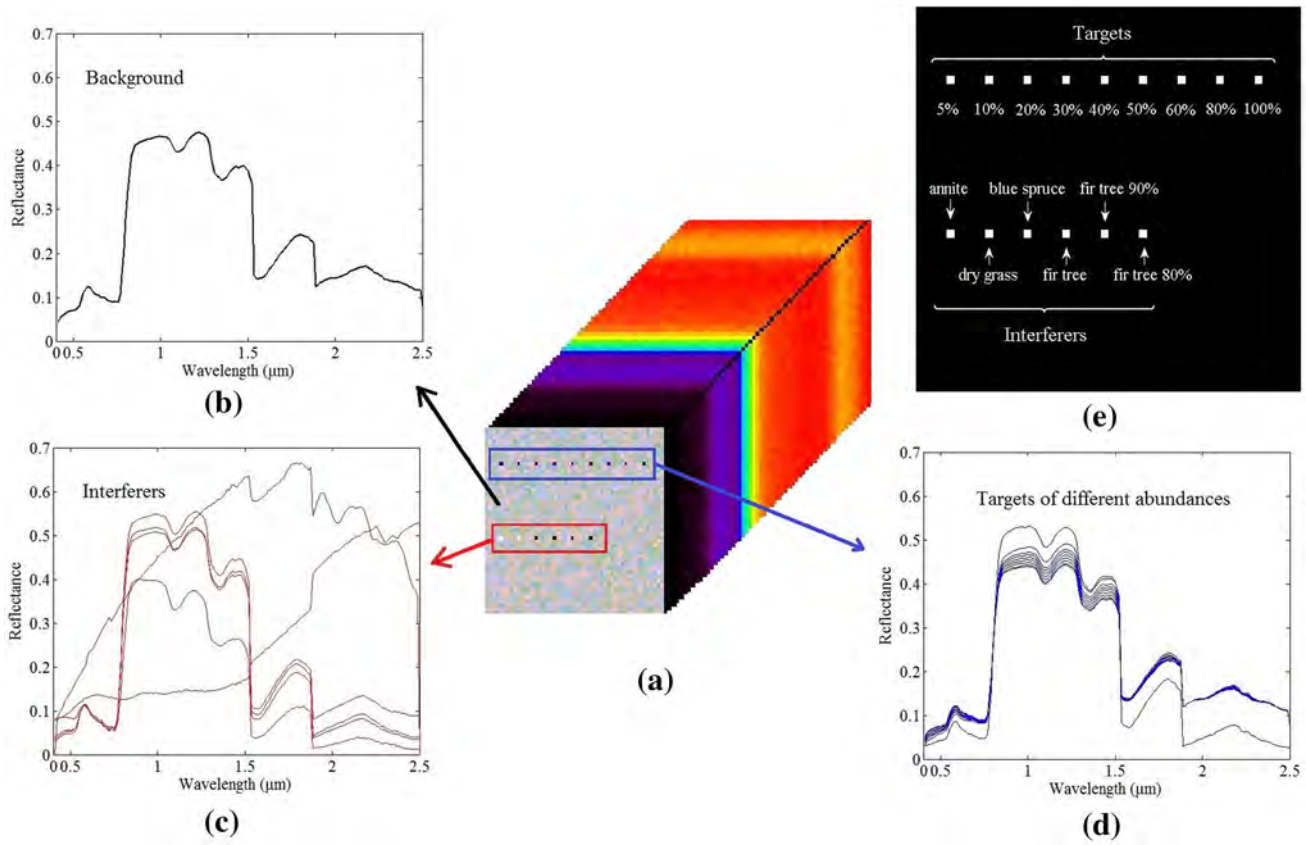



**Fig. 5** **a** 3D data cube of the synthetic image; **b–d** spectra of background, interferers and targets, respectively; and **e** ground truth of the synthetic image

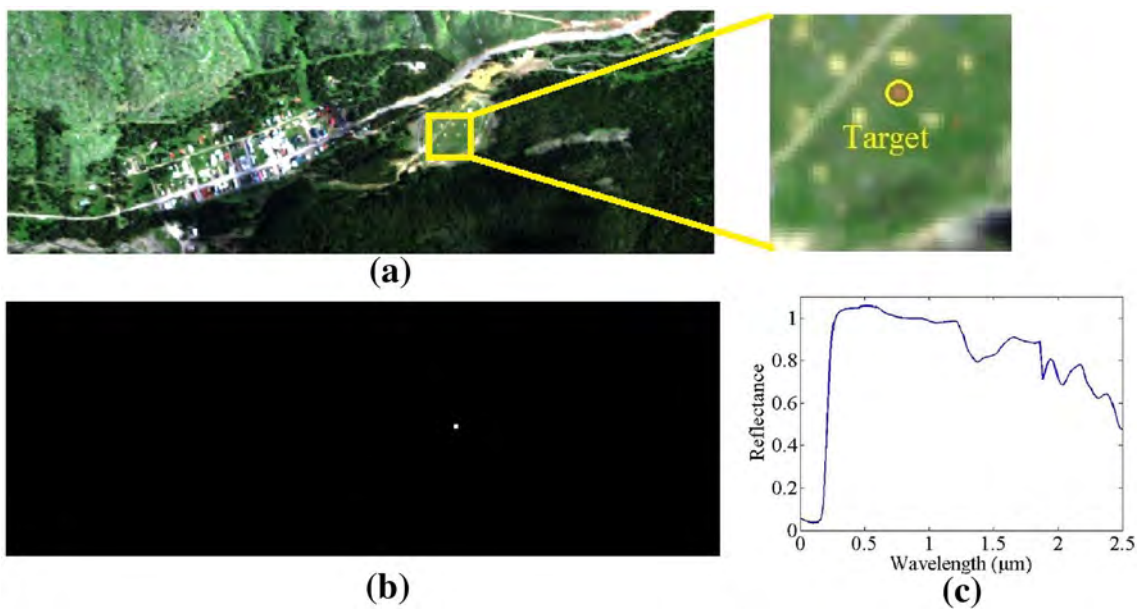



**Fig. 6** **a** HyMap image of Cook City in Montana and location of the target. **b** The ground truth; and **c** the target signature

curve that nearer to the upper left performs better, i.e., it provides higher TPR under the same FPR. The area under curve (AUC) provides a more visualized way to describe the ROC curve, and the larger AUC score means the better detection performance.

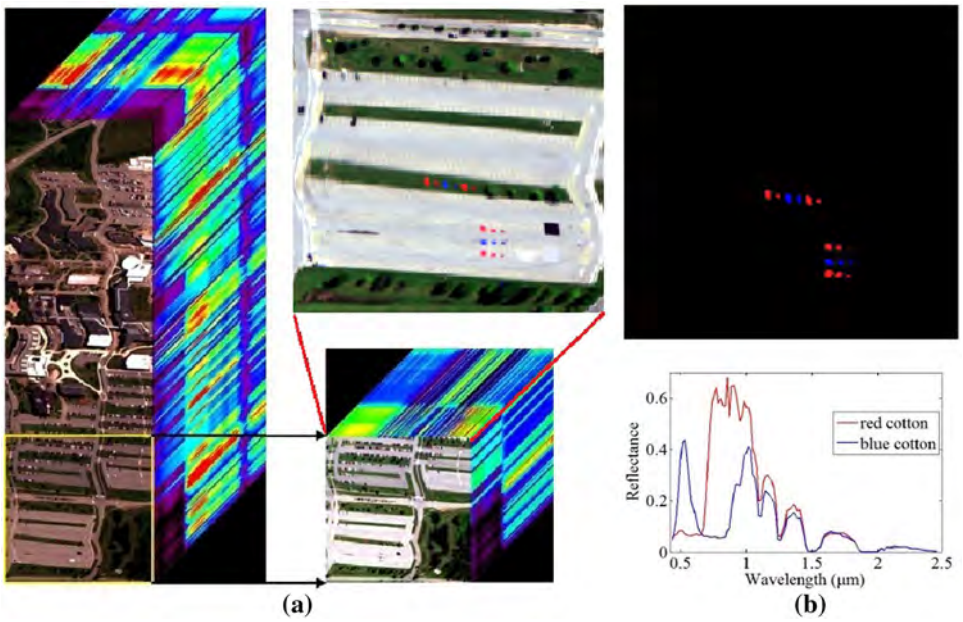



**Fig. 7** **a** The SpecTIR image over Rochester Institute of Technology and **b** the ground truth in which red cotton is shown in red and blue cotton is shown in blue

2. *Precision* this metric is used to measure the ability of correctly detecting targets with few false alarms [40]. The metric only focuses on the positive values in detection. For target detection, a certain number of pixels are regarded as "predictive positive," while some of them are true predictive positive (namely real targets) and the others are false alarms. Precision is defined as the ratio between true predictive positive and total predictive positive. In another words, this metric indicates the ability of resisting interference.

ROC curves, AUC and precision focus on different aspects and together provide a much comprehensive assessment of the performance of hyperspectral target detection algorithms.

### 5.3 Target detection accuracy analysis

1. *Target detection using synthetic image* we perform CEM, ACE, AMF, UBETD and RT-UBETD on our synthetic image. The blackbrush is set as the target of interest. The number of endmembers estimated by HySime is 5, so the number of background signatures is 4. Figure 8 shows the endmember extraction result obtained by RT-MSVA, it can be seen that RT-MSVA successfully extracts the five spectral signatures that form the synthetic image. For directly comparing these 5 algorithms, the results have been normalized to [− 1, 1]. Figure 9 shows the target detection scores for all the pixels of these 5 algorithms. The triangle markers represent the 9 target pixels inserted by the target signature of different abundances, while the circular markers represent the 6 interferers. The detection scores of UBETD and RT-UBETD are exactly the same and are nearly proportional to the abundance of target signature in the pixel. The other 3 algorithms get relatively good detection scores for targets with high abundances, but fail to detection deeply mixed targets. Besides, CEM, ACE and AMF all have false alarms on interferers. For better comparing the results, Fig. 10 shows the resulting binary images when the detection threshold is set as 0.05, 0.1, 0.2, 0.5 and 0.6, respectively. UBETD and RT-UBETD have no false alarm and can correctly detect all the targets under the threshold of 0.05. CEM and AMF both have false alarms, and the number of correctly detected targets for each algorithm is also less than UBETD and RT-UBETD. ACE can detect the most targets under the

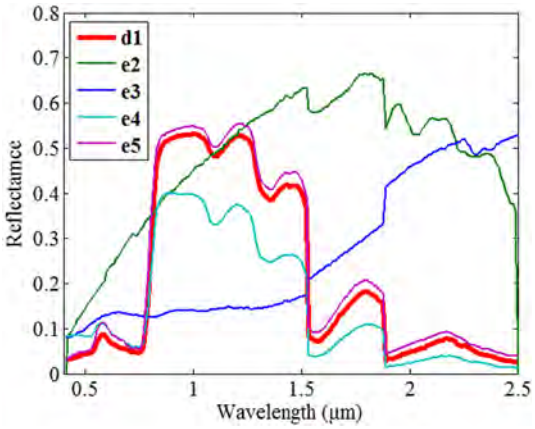



**Fig. 8** Endmember extraction results obtained by RT-MSVA in the synthetic image. In this figure, **d** denotes target and **e** denotes the following extracted endmember, i.e., background signature

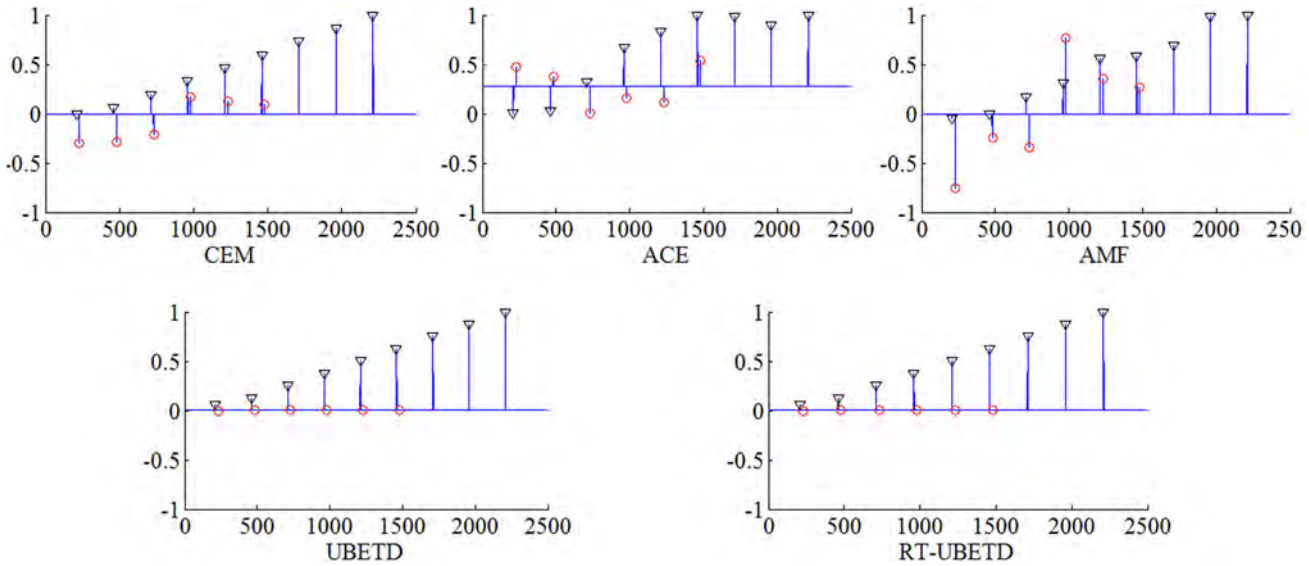
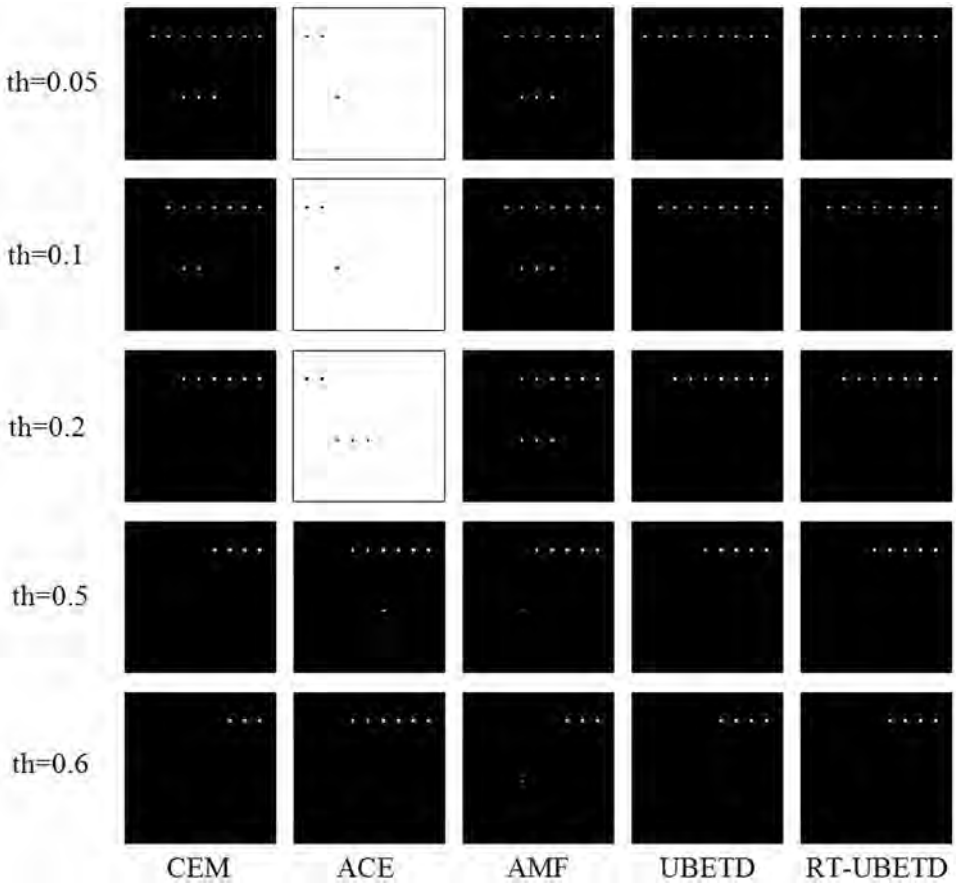



**Fig. 9** Target detection scores of all the algorithms. The triangle markers represent the targets of interest, while the circular markers represent the interferers

**Fig. 10** Results for the synthetic image under the threshold of 0.05, 0.1, 0.2, 0.5 and 0.6, respectively




threshold of 0.6, but also has false alarms for the other situations. Especially, since the background scores are relatively large for ACE, it fails to detect any target under the threshold of 0.05, 0.1 and 0.2. In general, the experiments on the synthetic image prove that UBETD and RT-UBETD perform much better than CEM, ACE and AMF in detecting targets of low abundance and inhibiting interferers.

2. *Target detection using real hyperspectral images* Fig. 11 shows the endmember extraction results of the HyMap image, in which the number of endmembers is estimated as 18. Figure 12 shows the detection results. We can see from the figures that all the 5 algorithms successfully detect the target desired near the center of the image in the red circle, where there are some pixels with comparatively large value. But the detection result of CEM and AMF also show a high brightness point in urban region by the left side of the image, which is a false alarm caused by an interferer. In fact, the detection scores of this interferer are even higher than the target, which means that if an inappropriate value is set as the threshold, the only predictive target achieved may be an interferer in reality. Because the fixed-point formats used to represent the intermediate variables have enough fractional bits, so UBETD and RT-UBETD can get the same detection results and share the same ROC curve and precision curve. Figure 13 shows the ROC curves and the precision curves, respectively. As for the number of target pixels and interferer pixels is quite small compared to the total number of pixels, the ROC curves are much similar to each other, and the AUC of UBETD and RT-UBETD (shown in Table 1) is slightly better than the others. On the other hand, the precision curves of these 5 algorithms show great differences. When the threshold is low, all these algorithms get some false alarms, so the precision is low too. But when the threshold increases to above 0.5, the precision of UBETD, RT-UBETD and ACE rises rapidly to 1, while the precision of CEM and AMF declines to 0. This is because the interferer gets the highest score for these 2 algorithms, so the number of correctly detected targets is 0 for high threshold. In general, UBETD and RT-UBETD successfully inhibit the interference and achieves an outstanding performance both in ROC curves and precision.

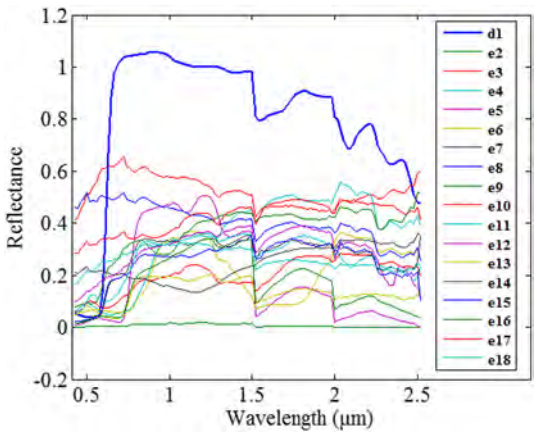



**Fig. 11** Endmember extraction results obtained by RT-MSVA in the HyMap hyperspectral image

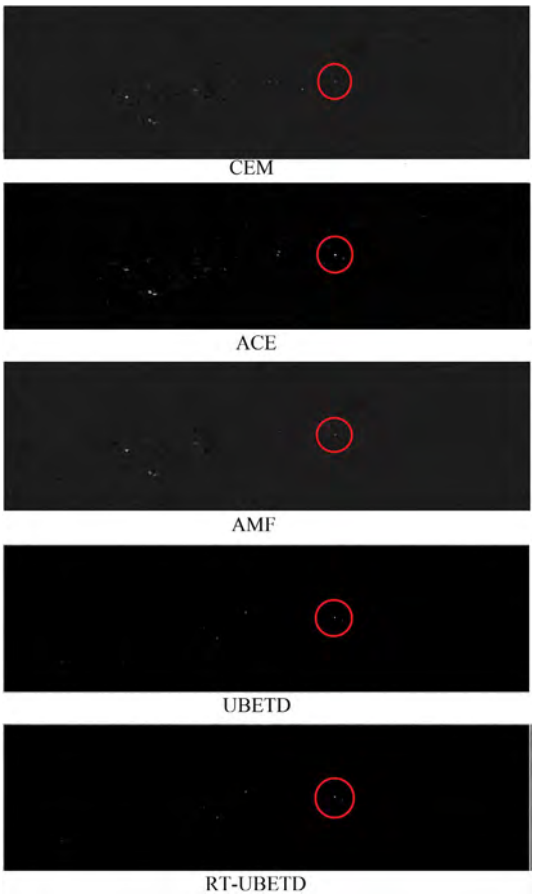



**Fig. 12** Detection results obtained by all the algorithms in the HyMap hyperspectral image

Figure 14 shows the endmember extraction results of SpecTIR data, in which the number of endmembers is estimated as 10. The detection results are shown in Fig. 15. There are two kinds of cotton artificially laid in this scene, and we take the red cotton as the target. As shown in Fig. 7b, there are two group of red cotton in the middle of the image with two for each group, and two rows in the bottom right with three for each row. UBETD and RT-UBETD detect all the red cotton without blue cotton. CEM and AMF detect some of the pixels that covered by blue cotton as predictive target, while ACE only detects part of the targets. Figure 16 shows the ROC curves and precision curves of these algorithms. It should be noted that the horizontal axis of ROC curves is shown in logarithmic form, because the large amount of total pixels makes the FPR quite small, so the ROC curves of these algorithms can show obvious difference in $10^{-4}$ level. The results show that UBETD and RT-UBETD perform better on both ROC curves and precision curves. The AUC of all the algorithms are also shown in Table 1.

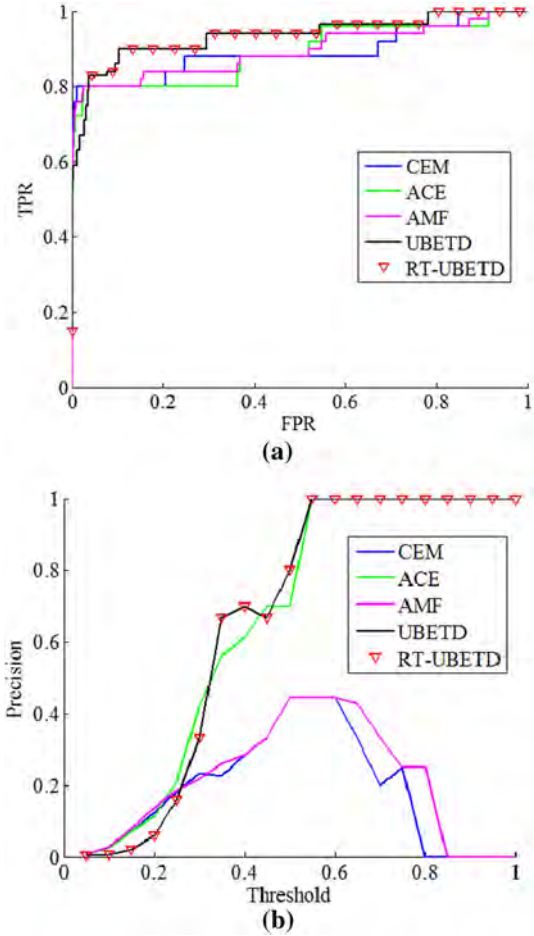



**Fig. 13** **a** ROC curves and **b** precision curves obtained by all the algorithms in the HyMap hyperspectral image

**Table 1** AUC obtained by all the algorithms in two real hyperspectral images

| | CEM | ACE | AMF | UBETD | RT-UBETD |
|---|---|---|---|---|---|
| HyMap | 0.8567 | 0.8883 | 0.8984 | 0.9280 | 0.9280 |
| SpecTIR | 1 | 1 | 1 | 1 | 1 |

3. *Multi-target detection using SpecTIR hyperspectral image* the multi-target detection experiment is performed on the SpecTIR image. Both red cotton and blue cotton are regarded as targets of interest. Figure 17a shows the endmember extraction results of SpecTIR data for multi-target detection. From the detection result shown in Fig. 17b, we can see that both of these two kinds of targets have been successfully detected. Figure 18 shows the ROC curve and the precision curve. Evidently, the proposed UBETD also has very good performance when detecting several targets simultaneously.

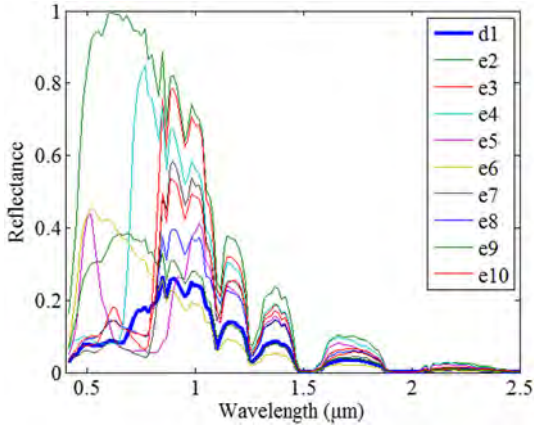



**Fig. 14** Endmember extraction results obtained by RT-MSVA in the SpecTIR hyperspectral image

## 5.4 Computational performance evaluation

We will analyze the computational performance of our FPGA implementation of RT-UBETD using the HyMap data set. Table 2 shows the summary of resource utilization. In our pre-analysis, the computational complexity of background extraction stage is much higher than target detection stage, so we implement higher parallelism in background extraction. For this reason, the resource utilization of background extraction is comparatively much higher than the target detection module. The image data are stored in a SDRAM, and the FPGA only reads one row at each time, so the utilization of block RAM and FIFO is comparatively low. On the other side, because of the computational complexity, the utilization of LUTs and DSP48E1s is high. Therefore, the proposed FPGA implementation is compute-bound. The maximum frequency of the whole system is 42.320 MHz.

Table 3 shows the processing times of RT-UBETD on our FPGA implementation and the equivalent software version in MATLAB. For RT-UBETD, MATLAB used 244.72 s to obtain target detection results, and our FPGA implementation used $2.84 \times 10^8$ clock periods. The clock frequency is 42.320 MHz, and the processing time is just 6.719 s. The endmember extraction module should traverse all the pixels for $n_U$ times, and the computational complexity is also larger than the target detection module, so it takes much longer time. Compared with software version in MATLAB, the FPGA implementation can achieve a 36 times speedup for this data set. As is introduced in [41], the HyMap sensor can collect 12–16 rows per second, and each row contains 512 pixels with 224 bands. The sensor generates data at approximately 2.5 MB per second. The size of the HyMap data set in our experiment is about 53.5 MB, so the sensor requires about 21.4 s to finish the data collection. In this way, our FPGA implementation of RT-

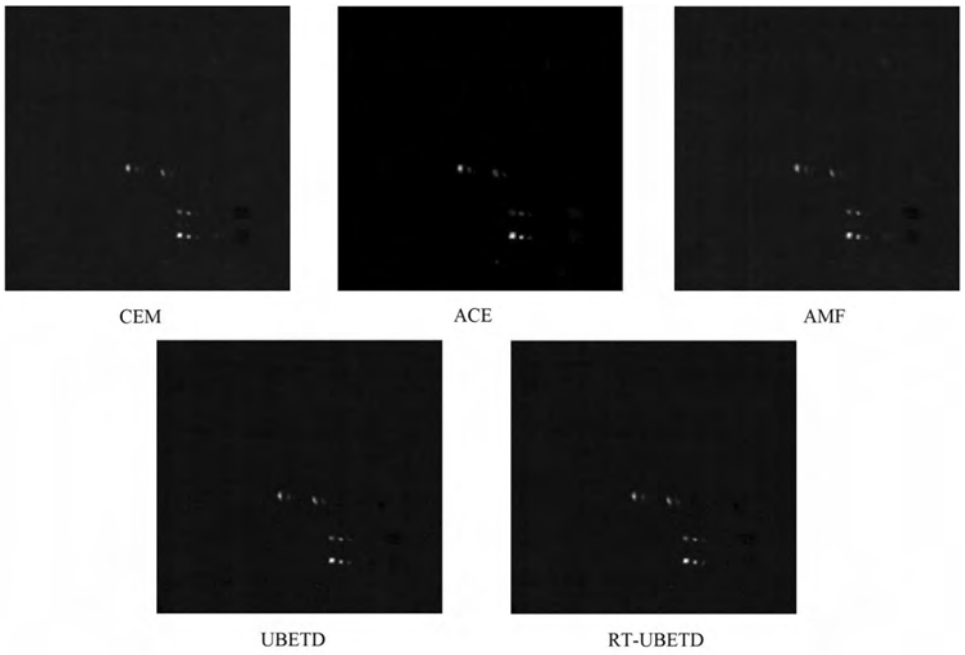



**Fig. 15** Detection results obtained by all the algorithms in the SpecTIR hyperspectral image

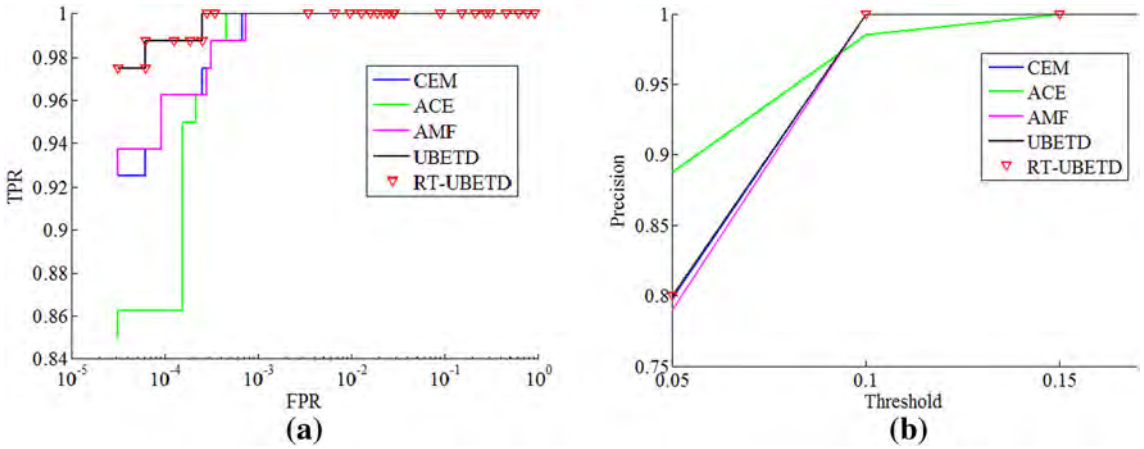



**Fig. 16** **a** ROC curves and **b** precision curves obtained by all the algorithms in the SpecTIR hyperspectral image

UBETD strictly meets the real-time requirements of sensors such as HyMap. We use the Xilinx Power Estimator (XPE) to do quick power estimation, and the estimated total on-chip power is 3.462 W. It should be noted that the actual power consumption may be lower than that. Furthermore, our implementation could be further optimized, as the current version has too much logic levels in some operations that brings limit to the system frequency.

## 6 Conclusions and future lines

An unsupervised background extraction-based target detection method is presented in this paper. The proposed method UBETD takes advantage of endmember extraction, which can automatically extract the representative material signatures from a hyperspectral image. This method has universal applicability that any endmember extraction algorithm can be applied to extract the background. This approach turns a "known target, unknown background" problem to an easier "known target, known background" problem, which is of great significance for hyperspectral target detection. The experiments have been conducted using a self-made synthetic image and two classic real hyperspectral data sets. The results show that UBETD and its hardware implementation RT-UBETD can achieve better performance in target detection and are particularly prominent in inhibiting interference in the background.

Also, an FPGA-based implementation is developed to meet the fast growing demand for real-time processing.

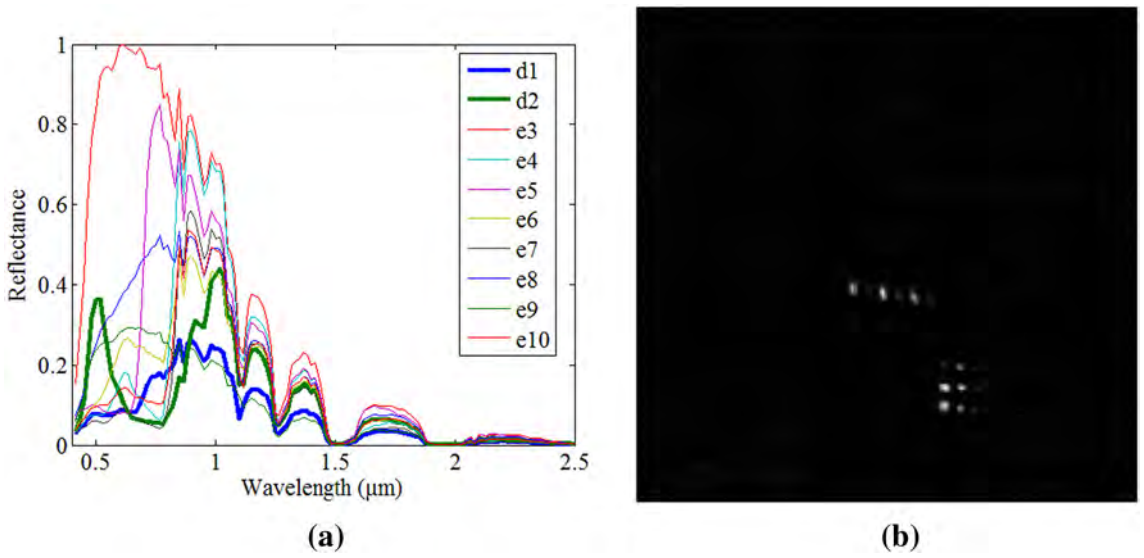



**Fig. 17** **a** Endmember extraction results and **b** target detection results obtained by RT-MSVA in the SpecTIR hyperspectral image for multi-target detection

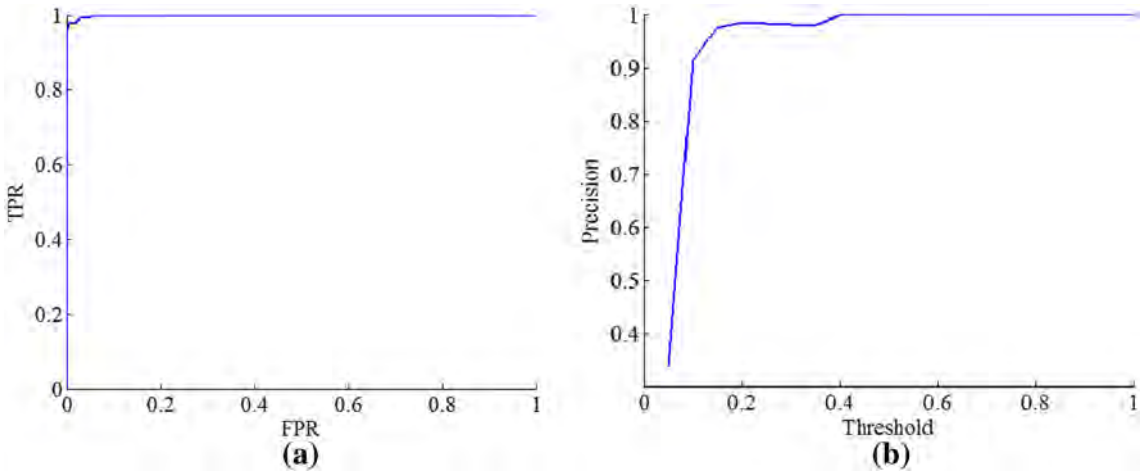



**Fig. 18** **a** ROC curve; and **b** precision curve of multi-target detection obtained by UBETD in the SpecTIR hyperspectral image

**Table 2** Summary of resource utilization of the FPGA implementation of RT-UBETD for the HyMap data

| Logic utilization | Available | Background extraction | | Target detection | | Total |
|---|---|---|---|---|---|---|
| | | Used | Utilization (%) | Used | Utilization | |
| Number of slice registers | 866,400 | 185,126 | 21 | 27,856 | 3 | 24% |
| Number of slice LUTs | 433,200 | 180,838 | 41 | 120,268 | 27 | 68% |
| Number of block RAM/FIFO | 1470 | 489 | 33 | 165 | 11 | 44% |
| Number of BUFG/BUFGCTRLs | 32 | 1 | 3 | 1 | 3 | 6% |
| Number of DSP48E1s | 3600 | 1262 | 35 | 729 | 20 | 55% |
| Maximum frequency (MHz) | | 43.320 | | 50.370 | | 42.320 |

The hardware implementation is carried out in a Xilinx Virtex-7 FPGA, which provides the top performance in the mainstream devices nowadays. A block-iterative inverse algorithm for large-scale real symmetric matrixes is introduced in our implementation to simplify the computation of inverse matrix. The results confirm that FPGA-based implementation of RT-UBETD can fully achieve real-time processing capabilities. But there are still some other problems to be solved. Only RT-MSVA and TCIMF have been applied in the UBETD method in this

**Table 3** Processing times measured for the proposed RT-UBETD and two specific modules with the HyMap image (MATLAB and FPGA implementation)

| | MATLAB (s) | FPGA | | Speedup |
|---|---|---|---|---|
| | | Processing time (s) | Clock periods | |
| Background extraction module (RT-MSVA) | 238.31 | 6.473 | $2.74 \times 10^8$ | 37× |
| Target detection module | 6.41 | 0.246 | $0.10 \times 10^8$ | 26× |
| RT-UBETD | 244.72 | 6.719 | $2.84 \times 10^8$ | 36× |

paper, and there is no comparison with other endmember extraction algorithms or target detection algorithms temporarily. In the future, we will continue this research to test other algorithms under our proposed method. Also, our hardware implementation will be further optimized. For example, the clock frequency may be increased by further pipelining the processing modules under the limit of computing resources, and the applicable ability should be developed by realizing real-time processing for images of larger size.

**Acknowledgements** This work was supported by the National Natural Science Foundation of China under Grant Nos. 91638201, 41722108, 41325004 and 41571349.

## References

1. Goetz, A.F.H., Vane, G., Solomon, J.E., Rock, B.N.: Imaging spectrometry for earth remote sensing. Science **228**(4704), 1147–1153 (1985)
2. Manolakis, D., Shaw, G.: Detection algorithms for hyperspectral imaging applications. IEEE Signal Process. Mag. **19**(1), 29–43 (2002)
3. Nasrabadi, N.M.: Hyperspectral target detection: an overview of current and future challenges. IEEE Signal Process. Mag. **31**(1), 34–44 (2013)
4. Eismann, M.T., Stocker, A.D., Nasrabadi, N.M.: Automated hyperspectral cueing for civilian search and rescue. Proc. IEEE **97**(6), 1031–1055 (2009)
5. Qian, Y., Yao, F., Jia, S.: Band selection for hyperspectral imagery using affinity propagation. IET Comput. Vision **3**(4), 213–222 (2009)
6. Dias, J.M.B., Plaza, A., Dobigeon, N., Parente, M., et al.: Hyperspectral unmixing overview: geometrical, statistical, and sparse regression-based approaches. IEEE J. Sel. Topics Appl. Earth Obs. Remote Sens. **5**(2), 354–379 (2012)
7. Chang, C.I., Sun, T.L., Althouse, M.L.G.: An unsupervised interference rejection approach to target detection and classification for hyperspectral imagery. Opt. Eng. **37**, 735–743 (1998)
8. Du, Q., Ren, H.: On the performance of CEM and TCIMF. Proc. SPIE Int. Soc. Opt. Eng. **5806**, 861–868 (2005)
9. Matteoli, S., Diani, M., Corsini, G.: A tutorial overview of anomaly detection in hyperspectral images. IEEE Aerosp. Electron. Syst. Mag. **25**(7), 5–27 (2010)
10. Stein, D.W.J., et al.: Anomaly detection from hyperspectral imagery. IEEE Signal Process. Mag. **19**(1), 58–69 (2002)
11. Kraut, S., Scharf, L.L., Butler, R.W.: The adaptive coherence estimator: a uniformly-most-powerful-invariant adaptive detection statistic. IEEE Trans. Signal Process. **53**(2), 417–438 (2005)
12. Robey, F.C., Fuhrmann, D.R., Kelly, E.J., Nitzberg, R.: ACFAR adaptive matched filter detector. IEEE Trans. Aerosp. Electron. Syst. **28**(1), 208–216 (1992)
13. Chang, C.I.: Target signature-constrained mixed pixel classification for hyperspectral imagery. IEEE Trans. Geosci. Remote Sens. **40**(5), 1065–1081 (2002)
14. Ren, H., Chang, C.I.: A target-constrained interference-minimized filter for subpixel target detection in hyperspectral imagery. In: Proceedings of IEEE 2000 International on Geoscience and Remote Sensing Symposium. IGARSS 2000. IEEE, vol. 4, pp. 1545–1547 (2000)
15. Zhang, B., et al.: Real-time target detection in hyperspectral images based on spatial-spectral information extraction. EURASIP J. Adv. Signal Process. **1**, 142 (2012)
16. Gomz, R.B., Lewis, A.J.: On-board processing for spectral remote sensing. In: ISPRS Special Session Future Intelligent Earth Observing Satellites (FIEOS'02) (2002)
17. Plaza, A., Plaza, J., Paz, A., Sánchez, S.: Parallel hyperspectral image and signal processing. IEEE Signal Process. Mag. **28**(3), 119–126 (2011)
18. Bing, Z.: Intelligent remote sensing satellite system. J. Remote Sens. **15**(3), 423–431 (2011)
19. Sánchez, S., Paz, A., Martin, G., Plaza, A.: Parallel unmixing of remotely sensed hyperspectral images on commodity graphics processing units. Concurr. Comput. Pract. Exp. **23**(13), 1538–1557 (2011)
20. Lysaght, P., Blodget, B., Mason, J., Young, J., Bridgford, B.: Enhanced architectures, design methodologies and CAD tools for dynamic reconfiguration of Xilinx FPGAs. In: Proceedings of International Conference Field Programmable Logic Applications, pp. 1–6 (2006)
21. Compton, K., Hauck, S.: Reconfigurable computing: a survey of systems and software. ACM Comput. Surv. **34**, 171–210 (2002)
22. Tessier, R., Burleson, W.: Reconfigurable computing for digital signal processing: a survey. J. VLSI Signal Process. Syst. **28**(1), 7–27 (2001)
23. Hauck, S.: The roles of FPGAs in reprogrammable systems. Proc. IEEE **86**(4), 615–639 (1998)
24. Plaza, A., Du, Q., Chang, Y.L., King, R.L.: High performance computing for hyperspectral remote sensing. IEEE J. Sel. Topics Appl. Earth Obs. Remote Sens. **4**(3), 528–544 (2011)
25. Gonzalez, C., Lopez, S., Mozos, D., et al.: A novel FPGA-based architecture for the estimation of the virtual dimensionality in remotely sensed hyperspectral images. J. Real Time Image Proc. **43**(5), 1–12 (2015)
26. Wang, J., Chang, C., Cao, M.: FPGA design for constrained energy minimization. In: Optical Technologies for Industrial, Environmental, and Biological Sensing. International Society for Optics and Photonics, pp. 262–273 (2004)
27. Yang, B., Yang, M., Plaza, A., et al.: Dual-mode FPGA implementation of target and anomaly detection algorithms for real-time hyperspectral imaging. IEEE J. Sel. Topics Appl. Earth Obs. Remote Sens. **8**(6), 2950–2961 (2015)

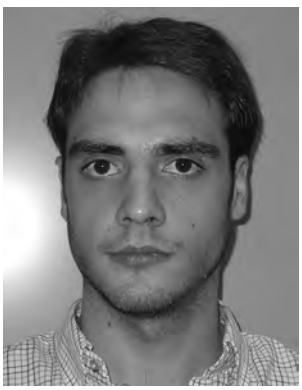
28. Gonzalez, C., Bernabe, S., Mozos, D., et al.: FPGA implementation of an algorithm for automatically detecting targets in remotely sensed hyperspectral images. IEEE J. Sel. Topics Appl. Earth Obs. Remote Sens. **9**(9), 1–10 (2016)
29. Bernabé, S., López, S., Plaza, A., et al.: FPGA design of an automatic target generation process for hyperspectral image analysis. In: IEEE, International Conference on Parallel and Distributed Systems, IEEE Computer Society, pp. 1010–1015 (2011)
30. Li, C., Gao, L., Plaza, A., Zhang, B.: FPGA implementation of a maximum simplex volume algorithm for endmember extraction from remotely sensed hyperspectral images. J. Real Time Image Process. **1**, 1–14 (2017)
31. Chang, C.I.: Target abundance-constrained subpixel detection: partially constrained least-squares methods. In: Hyperspectral Imaging: Techniques for Spectral Detection and Classification, pp. 39–41. Kluwer, New York, NY (2003)
32. Winter, M.E.: N-FINDR: an algorithm for fast autonomous spectral endmember determination in hyperspectral data. Proc. SPIE **3753**, 266–275 (1999)
33. Chang, C.I., Wu, C., Liu, W., Ouyang, Y.C.: A growing method for simplex-based endmember extraction algorithms. IEEE Trans. Geosci. Remote Sens. **44**(10), 2804–2819 (2006)
34. Dias, J.M.B., Nascimento, J.M.P.: Hyperspectral subspace identification. IEEE Trans. Geosci. Remote Sens. **46**(8), 2435–2445 (2008)
35. Gonzalez, C., et al.: FPGA implementation of the HySime algorithm for the determination of the number of endmembers in hyperspectral data. IEEE J. Sel. Topics Appl. Earth Obs. Remote Sens. **8**(6), 1–14 (2015)
36. Thambi, M.C., Dharan, N.B., Rajaram, S.: FPGA implementation of low cost matrix inverse computation for MIMO systems. Int. J. Appl. Eng. Res. **10**(20), 15622–15626 (2015)
37. Tao, L.I., Zhang, Z.P.: Matrix inversion by FPGA. Commun. Technol. **43**(11), 147–149 (2010)
38. Zhang, G., Shen, H., Shi, F., Huo, Y.: Block iterative inverse algorithm for a large-scale real matrix. Wirel. Internet Technol. **6**, 127–129 (2015)
39. Eftestøl, T.: Controlling true positive rate in ROC analysis. Comput. Cardiol. Conf. **373**(42), 353–356 (2009)
40. Stock, E.M., et al.: Estimation of disease prevalence, true positive rate, and false positive rate of two screening tests when disease verification is applied on only screen-positives: a hierarchical model using multi-center data. Cancer Epidemiol. **36**(36), 153–160 (2012)
41. Cocks, T., Jenssen, R., Stewart, A., Wilson, I., Shields, T.: The HyMap airborne hyperspectral sensor: the system, calibration and performance. In Proceedings of 1st EARSEL Workshop on Imaging Spectroscopy, Zurich, Switzerland, pp. 37–42 (1998)


**Cong Li** received his B.S. degree in communication engineering from Nanjing University, Nanjing, China, in 2013. He is currently pursuing the Ph.D. degree in the Institute of Remote Sensing and Digital Earth, Chinese Academy of Science, Beijing, China. His research interests include hyperspectral remote sensing, hyperspectral unmixing, hyperspectral image processing and high-performance computing with FPGAs.

**Lianru Gao** received the B.S. degree in civil engineering from Tsinghua University, Beijing, China, in 2002, and the Ph.D. degree in cartography and geographic information system from Institute of Remote Sensing Applications, Chinese Academy of Sciences, Beijing, China, in 2007. He is currently a Professor with the Key Laboratory of Digital Earth Science, Institute of Remote Sensing and Digital Earth, Chinese Academy of Sciences. He has published over 90 papers in China and abroad. His research interests include hyperspectral image processing and information extraction. He is a recipient of the recognition of Best Reviewers of the IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing in 2015.

**Yuanfeng Wu** received the B.S. and M.S. degrees in Computer Science from China University of Mining and Technology (Beijing), China, the Ph.D. degree in Cartography and Geographical Information System from Graduate University of Chinese Academy of Sciences, China, in 2010. He is currently an Associate Professor with the Key Laboratory of Digital Earth Science, Institute of Remote Sensing and Digital Earth, Chinese Academy of Sciences. His research interests include the development of high-performance computing algorithms and computer software systems in hyperspectral image processing.

**Bing Zhang** received the B.S. degree in geography from Peking University, Beijing, China, the M.S. and Ph.D. degrees in remote sensing from the Institute of Remote Sensing Applications, Chinese Academy of Sciences (CAS), Beijing, China. Currently, he is a Full Professor and the Deputy Director of the Institute of Remote Sensing and Digital Earth (RADI), Chinese Academy of Sciences (CAS), where he has been leading key scientific projects in the area of hyperspectral remote sensing for more than 20 years. His research interests include the development of Mathematical and Physical models and image processing software for the analysis of hyperspectral remote sensing data in many different areas, such as geology, hydrology, ecology and botany. He has developed 5 software systems in the image processing and applications. His creative achievements were rewarded 8 important prizes from Chinese government, and special government allowances of the Chinese State Council. He was awarded the National Science Foundation for Distinguished Young Scholars of China in 2013. He was also awarded as one of the top four scientists among more than 100 professors in RADI in 2015 by Chinese Academy of Sciences. He has authored more than 300 publications, including 171 journal papers. He has edited 6 books/contributed book chapters on hyperspectral image processing and subsequent applications, and those books serve as the main materials for education and research in hyperspectral remote sensing in China. He has been a professor of University of Chinese Academy of Sciences and gave lessons of Hyperspectral Remote Sensing for more than 10 years. He has been the advisor of 29 Ph.D. dissertations and more than 12 M.S. dissertations. He is currently serving as the Associate Editor for IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing (IEEE JSTARS) since 2011, and he is also the guest editor for IEEE JSTARS special issue on "Hyperspectral Remote Sensing: Theory, Methods, and Applications" in 2013, IEEE JSTARS special issue on "Big Data in Remote Sensing" in 2015 and Pattern Recognition Letters (PRL) special issue on "Advances in Pattern Recognition in Remote Sensing" in 2015. He has served as Technical Committee Member of IEEE Workshop on Hyperspectral Image and Signal Processing (IEEE WHISPERS) since 2011, and as the Chair of China National Committee of International Society for Digital Earth since 2012. He is the Student Paper Competition Committee member in IGARSS 2015 and 2016, and Scientific Committee Member of IGRASS in 2014.



**Javier Plaza** received the M.Sc. degree in computer science from the University of Extremadura, Cáceres, Spain, in 2002, where he is currently pursuing the Ph.D. degree. He is currently an Assistant Professor in the Computer Science Department, University of Extremadura. His current research work is focused on the development of efficient implementations of nonlinear mixture model-based algorithms for abundance estimation of materials in hyperspectral scenes. He is also involved in the design and configuration of commodity cluster computing architectures for high-performance hyperspectral analysis. Other major research interests include development of quantitative and comparative applications for remote sensing, and the design of web-based applications for hyperspectral image processing.

**Antonio Plaza** was born in Caceres, Spain, in 1975. He is an Associate Professor (with accreditation for Full Professor) with the Department of Technology of Computers and Communications, University of Extremadura, where he is the Head of the Hyperspectral Computing Laboratory (HyperComp), one of the most productive research groups working on remotely sensed hyperspectral data processing worldwide. His main research interests comprise hyperspectral data processing and parallel computing of remote sensing data. He has been the advisor of 12 Ph.D. dissertations and more than 30 M.Sc. dissertations. He was the Coordinator of the Hyperspectral Imaging Network, a European project with total funding of 2.8 million Euro. He has authored more than 500 publications, including 152 journal papers (more than 100 in IEEE journals), 22 book chapters and over 240 peer-reviewed conference proceeding papers (94 in IEEE conferences). He has edited a book on High-Performance Computing in Remote Sensing for CRC Press/Taylor and Francis and guest edited 9 special issues on hyperspectral remote sensing for different journals. Dr. Plaza is a Fellow of IEEE "for contributions to hyperspectral data processing and parallel computing of Earth observation data." He is a recipient of the recognition of Best Reviewers of the IEEE Geoscience and Remote Sensing Letters (in 2009) and a recipient of the recognition of Best Reviewers of the IEEE Transactions on Geoscience and Remote Sensing (in 2010), a journal for which he served as Associate Editor in 2007–2012. He is also an Associate Editor for IEEE Access and was a member of the Editorial Board of the IEEE Geoscience and Remote Sensing Newsletter (2011–2012) and the IEEE Geoscience and Remote Sensing Magazine (2013). He was also a member of the steering committee of the IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing (JSTARS). He is a recipient of the 2013 Best Paper Award of the JSTARS journal, and a recipient of the most highly cited paper (2005–2010) in the Journal of Parallel and Distributed Computing. He received the best paper awards at the IEEE International Conference on Space Technology and the IEEE Symposium on Signal Processing and Information Technology. He is a recipient of the Best Ph.D. Dissertation award at University of Extremadura, a recognition also received by six of his Ph.D. students. He served as the Director of Education Activities for the IEEE Geoscience and Remote Sensing Society (GRSS) in 2011–2012 and is currently serving as President of the Spanish Chapter of IEEE GRSS (since November 2012). He has served as a proposal evaluator for the European Commission, the National Science Foundation, the European Space Agency, the Belgium Science Policy, the Israel Science Foundation and the Spanish Ministry of Science and Innovation. He has reviewed more than 500 manuscripts for over 50 different journals. He is currently serving as the Editor-in-Chief of the IEEE Transactions on Geoscience and Remote Sensing journal. Additional information: http://www.umbc.edu/rssipl/people/aplaza.