Fast Real-Time Causal Linewise Progressive Hyperspectral Anomaly Detection via Cholesky Decomposition

Lifu Zhang , *Senior Member, IEEE*, Bo Peng , *Student Member, IEEE*, Feizhou Zhang, Lizhe Wang, Hongming Zhang , Peng Zhang, and Qingxi Tong

Abstract—**Real-time processing of anomaly detection has become one of the most important issues in hyperspectral remote sensing. Due to the fact that most widely used hyperspectral imaging spectrometers work in a pushbroom fashion, it is necessary to process the incoming data line in a causal linewise progressive manner with no future data involved. In this study, we proposed several processes to well improve the computational performance of real-time causal linewise progressive anomaly detection (RCLP-AD). At first, Cholesky decomposition along with linear system solving (CDLSS) was used since the background statistical matrix are symmetric positive definite. The computational performance as well as the numerical stabilities is well improved. In order to show the computational advantage of the proposed method, we did a comprehensive comparative analysis regarding the computational complexity of different linewise processing techniques, in terms of the theoretical floating point operations (flops) and the real computer processing time. Moreover, the symmetric property of some intermediate resulting matrices in the process is considered for further computational optimization. Finally, from an onboard detection point of view, we defined the line-varying global background (i.e., an area covered by recently acquired data lines) to improve the detection power. To substantiate the performance of the CDLSS-based RCLP-AD regarding the accuracy and efficiency, two hyperspectral datasets were used in our experiments.**

Index Terms—**Anomaly detection, hyperspectral, linewise, real time.**

I. INTRODUCTION

HYPERSPECTRAL data, with very high spectral resolution, are abundant among the spectral and radiometric information collected from observed objects [1]. To identify certain features of objects, various hyperspectral applications have been developed, including mineral mapping [2], image precision classification [3], and target or anomaly detection [4]. Anomaly detection using hyperspectral data has received growing interest from researchers worldwide [5]–[7]. It is an unsupervised process with no *a priori* knowledge of data needed. Generally the anomaly is defined as a pixel or sample, of which the spectral signature is distinct from that of the surrounding background materials. Anomalies usually contain very useful information, such as crop diseases and hot spot fires, and typically behave similarly to the background in multispectral or panchromatic images. By taking advantage of the massive reflective and emissive information contained in hyperspectral data, some subtle differences between anomalies and the background can be revealed by hyperspectral anomaly detectors [8]–[12].

The Reed–Xiaoli detector (RXD), which was developed by Reed and Yu [8], is the benchmark anomaly detector. It calculates the Mahalanobis distance between the pixel under test (PUT) and the background. Generally, the background statistics are represented by covariance or correlation matrix of all samples. For better detection performance, a number of modified variants of RXD have been developed, including the uniform target detector (UTD) and low probability detection (LPD) [9], random-selection-based anomaly detector (RSAD) [10], and the kernel RX-algorithm [13]. UTD and LPD use the unity vector (all components are equal to one) as the matched signal, whereas RXD uses the PUT. In RSAD, to prevent the background from being contaminated by anomalies, randomly selected pixels are used to calculate the sample correlation matrix, allowing for the low probability of the occurrence being an anomaly. The Kernel RX was developed via a nonlinear function to map the original data into a high-dimensional feature space, exploiting the high-order nonlinear relationships between different spectral bands within targets or background samples.

However, conventional detection algorithms do not work well in situations that require real-time processing. In those circumstances, the data should be processed immediately once being collected without the use of any future data, and the output should be produced in negligible time for immediate decision making. For example, fire damage and moving targets need to be detected in real time when the data are acquired. Otherwise the fire would lead to great damage, and the moving targets or anomalies would disappear quickly after turning up. As demonstrated in [14], hyperspectral real-time anomaly detection has

L. Zhang, L. Wang, H. Zhang, and Q. Tong are with the Institute of Remote Sensing and Digital Earth, Chinese Academy of Sciences, Beijing 100101, China (e-mail: zhanglf@radi.ac.cn; wanglz@radi.ac.cn; zhanghm@radi.ac.cn; tqxi@263.net).

B. Peng and P. Zhang are with the Institute of Remote Sensing and Digital Earth, Chinese Academy of Sciences, Beijing 100101, China, and also with the University of Chinese of Academy of Sciences, Beijing 100049, China (e-mail: pengbo@radi.ac.cn; minand19@163.com).

F. Zhang is with the School of Earth and Spatial Sciences, Peking University, Beijing 100080, China (e-mail: zhangfz@pku.edu.cn).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/JSTARS.2017.2725382

been successfully applied to onboard crop pest and disease monitoring, which contributes to efficient and precision agriculture. Thus, it is necessary to develop fast, real-time anomaly detection algorithms.

In general, a heavy computational load results from background suppression. Computing the inverse of sample correlation matrix R or covariance matrix K could be interpreted as a whitening process to suppress the background [15], [16]. To speed up the entire process, parallel computing is widely used via high-performance architectures, such as multicore platforms [17], graphics processing units [18], multiple digital signal processors (multi-DSPs) [19], and field programmable gate arrays (FPGAs) [20]. These approaches take advantage of the numerous parallelizable processing units, as well as the data structures. Each data line or sample is processed simultaneously. In addition, to further accelerate the process, optimizations were proposed using QR decomposition and linear system solving [17]. However, detectors based on these techniques should be called fast, rather than real-time, detectors, because they process data based on the entire dataset. Real-time detectors should output detection results for the incoming data immediately without any future data.

To implement real-time detection, causal processing should be stressed at first. The term *causal* indicates that no future data can be used in the process [21]. The background statistics can only be updated using previous information and the data currently acquired. Second, the process should be conducted progressively along with the incoming data. Third, the computer processing time required for processing the incoming data should be nearly constant. It is not necessary to recalculate the background statistics with all of the samples in a global way when new data are acquired [22]. Causal progressive processing is very essential for onboard real-time detection, but has not been paid much attention to. From a data acquisition point of view, two major real-time processing fashions are involved. The first one is causal samplewise progressive process [15], [22]–[24] designed for processing data sample by sample corresponding to whiskbroom imaging spectrometers. The second one is causal linewise progressive process [23]–[25] mainly developed for processing data line by line associated with pushbroom sensors. Both of them take advantage of Woodbury's identity to update the suppressed background statistics, i.e., the inverse sample correlation/covariance matrix. It should be noted that the line-by-line architecture proposed in [26]–[28] is different from causal processing. It is a local detector, requiring future data relative to the PUT in the incoming data line to construct the local dual window. However, causal processing cannot get access to any future data since they have not been acquired [22], [29].

It is worth noting that the majority of current widely used imaging spectrometers collect data line by line, such as the spaceborne Hyperion and FTHSI, airborne HYDICE, ProSpecTIR VIS, ROSIS, HySpecx, CASI, AISA-FENIX, and PHI. Since conventional sequential execution of global real-time causal linewise progressive detection is still computationally expensive, faster processes are especially in need. More

importantly, Woodbury's identity currently in wide use can lead to severe numerical instabilities owing to roundoff error [30], [31]. Finally, due to the fact that onboard hyperspectral imagers will scan across a large scale of the ground, data acquired at the very beginning are in little relation to the incoming data. It is not appropriate to use the entire dataset for background suppression. As a result, to further cut down the time consumption and meanwhile improve the numerical stabilities for onboard real-time anomaly detection, we need to design faster and more stable real-time processing methods.

In this study, several processes are proposed for sequential execution of the real-time causal linewise progressive anomaly detector to reduce the computational complexity significantly while improving the detection performance. Our main contributions include:

1) We improved the computational performance of the real-time process significantly via Cholesky decomposition and linear system solving (CDLSS). The background suppression process is modified into a process to find the solution to the linear equations. Thus, the numerical stabilities and computing efficiency are both well improved. It is worth noting that the sample correlation or covariance matrix was proved to be symmetric positive definite, which has received little interests in real-time progressive anomaly detection. However, this issue is of great significance since it contributes to simplifying and accelerating the detection process while keeping the numerical stabilities and detection accuracy. By virtue of the symmetric property of some intermediate resulting matrices, the computation is further simplified. Also CDLSS have great sense in hardware parallel implementation.

2) We performed a comprehensive comparative analysis regarding the computational load of different real-time processing techniques, so that we are able to find the most efficient processing strategy for a specific pushbroom imaging spectrometer. Such a comparison has not been reported in the literature and showed the computational advantage of the method proposed in this work.

3) From an onboard detection point of view, we proposed to use the latest collected data lines for better background suppression and detecting weak anomalies. Those data lines, which are collected in the early stages, are removed from the background at the same time the new data line is acquired. Moreover, the process to discard old background information does not affect the computational performance but improve the detection accuracy.

The organization of this paper is as follows. Section II provides the brief description of causal linewise anomaly detection based on the well-known RXD. Section III details our proposed method for real-time causal linewise processing. In Section IV, computational performance of different processing strategies is discussed quantitatively and comparatively. Then, in Section V, we use a dataset collected by the ground-based pushbroom imaging spectrometer system in the laboratory and another airborne hyperspectral dataset to substantiate the performance of the proposed technique.

II. CAUSAL LINEWISE PROGRESSIVE ANOMALY DETECTION (CLP-AD)

The sample covariance matrix-based anomaly detection (K-AD), which was developed by Reed and Yu (RX)[8], is the benchmark anomaly detector. Numerous variants of the RX detector have been proposed, such as the sample correlation matrix based anomaly detection (R-AD) [9]. Let $\boldsymbol{r}_i = (r_{i1}, r_{i2}, \ldots, r_{iB})^T$ denotes the $B \times 1$ sample spectral vector, where B is the number of spectral bands.

A. K-AD and R-AD

The K-AD ($\delta^{K-AD}(\boldsymbol{r}_i)$), using a sample covariance matrix $\boldsymbol{K}$ of the background samples, is formulated as follows:

$$\delta^{K-AD}(\boldsymbol{r}_i) = (\boldsymbol{r}_i - \boldsymbol{\mu})^T \boldsymbol{K}^{-1} (\boldsymbol{r}_i - \boldsymbol{\mu}) \gtrless \eta \qquad (1)$$

where $\mu = (1/N) \sum_{i=1}^{N} \boldsymbol{r}_i$ is the $B \times 1$ global sample mean vector of the background, N is the total number of samples, and $\boldsymbol{K} = (1/N) \sum_{i=1}^{N} (\boldsymbol{r}_i - \boldsymbol{\mu})(\boldsymbol{r}_i - \boldsymbol{\mu})^T$. The inverse of $\boldsymbol{K}$, $\boldsymbol{K}^{-1}$, is used to suppress the image background [21]. $\delta^{K-AD}(\boldsymbol{r}_i)$ in (1) is interpreted as the Mahalanobis distance. If the output $\delta^{K-AD}(\boldsymbol{r}_i)$ is greater than the threshold η, $\boldsymbol{r}_i$ can be considered an anomaly; otherwise, it is considered part of the background.

A variant of K-AD, using a sample correlation matrix $\boldsymbol{R}$ referred to as R-AD ($\delta^{R-AD}(\boldsymbol{r}_i)$) developed in [9], is specified by

$$\delta^{R-AD}(\boldsymbol{r}_i) = \boldsymbol{r}_i^T \boldsymbol{R}^{-1} \boldsymbol{r}_i \gtrless \eta \qquad (2)$$

where $\boldsymbol{R} = (1/N) \sum_{i=1}^{N} \boldsymbol{r}_i \boldsymbol{r}_i^T$.

B. Causal Linewise Progressive Processing

To implement real-time onboard anomaly detection, the background statistics should be updated in a causal progressive manner. In this work, we mainly focused on linewise processing of R-AD, which exploits sample correlation matrix.

Assume $\boldsymbol{X}_i = (\boldsymbol{r}_1, \boldsymbol{r}_2, \ldots, \boldsymbol{r}_L)$ denotes a line of data, where L is the length of a data line, i.e., the number of samples or pixels in a data line. The term "causal" is emphasized since no future data to be visited can be used. Rewrite $\boldsymbol{R}$ as $\boldsymbol{R}(n)$, where $\boldsymbol{R}(n) = [1/(nL)] \sum_{i=1}^{n} \boldsymbol{X}_i \boldsymbol{X}_i^T$. This indicates that the sample correlation matrix is generated by acquired data lines from $\boldsymbol{X}_1$ to $\boldsymbol{X}_n$. Accordingly, CLP-AD is specified by

$$\delta^{\text{CLP-AD}}(n, i) = \boldsymbol{r}_i^T \boldsymbol{R}^{-1}(n-1) \boldsymbol{r}_i \qquad (3)$$

where $\boldsymbol{r}_i$ is the ith sample vector in the nth incoming data line $\boldsymbol{X}_n$. It should be noted that the use of $\boldsymbol{R}(n-1)$, instead of $\boldsymbol{R}(n)$, is to prevent $\boldsymbol{X}_n$ from being suppressed in the background [21]. Each time a new data line $\boldsymbol{X}_n$ was acquired, we used the $\boldsymbol{R}^{-1}(n-1)$ derived previously for detection at first, and then updated $\boldsymbol{R}(n-1)$ to $\boldsymbol{R}(n)$ with $\boldsymbol{X}_n$.

To avoid the singularity of $\boldsymbol{R}(n)$, especially the initial $\boldsymbol{R}(n)$ at the very beginning of the real-time process, the number of samples selected for calculating the initial $\boldsymbol{R}(n)$ should be at least greater than the dimensionality of $\boldsymbol{R}(n)$ (i.e., the number

of spectral bands B). Thus, the initial $\boldsymbol{R}(n)$ was calculated in a global way using the first k lines of data, which contain more than B samples. Here, $\boldsymbol{k} \geq \text{Int}(B/L) + 1$ so that $kL \geq B$. However, it is worth noting that the initial $\boldsymbol{R}(n)$ might be still ill-conditioned if the selected data are spectrally highly correlated even though enough samples were used. Accordingly, we added a small constant value λ to the diagonal entries of $\boldsymbol{R}(n)$ (i.e., $\boldsymbol{R}(n) := \boldsymbol{R}(n) + \lambda I_B$, where I_B is a B-dimensional identity matrix).

III. REAL-TIME CLP-AD (RCLP-AD)

For real-time implementation of each processing stage, Cholesky decomposition was conducted. The line-varying background in this study was developed to acquire better detection accuracy.

A. Cholesky Decomposition of RCLP-AD

Assuming x is a $B \times 1$ nonzero vector, we have

$$x^T \boldsymbol{R}(n-1)x = x^T \left[\frac{1}{(n-1)L} \sum_{i=1}^{n-1} \boldsymbol{X}_i \boldsymbol{X}_i^T \right] x$$

$$= \frac{1}{(n-1)L} \sum_{i=1}^{n-1} (\boldsymbol{X}_i^T x)^T (\boldsymbol{X}_i^T x)$$

$$= \frac{1}{(n-1)L} \sum_{i=1}^{n-1} \|\boldsymbol{X}_i^T x\|^2 \geq 0 \qquad (4)$$

which indicates that $\boldsymbol{R}(n-1)$ is positive semidefinite with all of the eigenvalues ($\lambda_1, \lambda_2, \ldots, \lambda_B$) being nonnegative

$$\lambda_i \geq 0 \quad (i = 1, 2, \ldots, B). \qquad (5)$$

In addition, to avoid the ill-conditioned problem, two steps are involved: 1) more than B samples are required to calculate $\boldsymbol{R}(n-1)$, and 2) a diagonal matrix with very small scale is added to $\boldsymbol{R}(n-1)$. Moreover, one of the prerequisites of RCLP-AD is that the background should be suppressed. In other words, only when $\boldsymbol{R}(n-1)$ is nonsingular can RCLP-AD keep working. Thus, none of λ_i ($i = 1, 2, \ldots, B$) could be equal to zero

$$\lambda_i > 0 \quad (i = 1, 2, \ldots, B). \qquad (6)$$

As a result, $\boldsymbol{R}(n-1)$ is a positive definite symmetric matrix. This characteristic is important because it can simplify the entire process while keeping the numerical stabilities.

Here, Cholesky decomposition would help speed up RCLP-AD significantly. A positive definite symmetric matrix can be factorized into a lower triangular matrix and its transpose via Cholesky decomposition. The Crout algorithm is used to perform this factorization process, which is computationally efficient and numerically stable without any pivoting at all as indicated by [32, Section 2.9]. We have

$$\boldsymbol{R}(n-1) = \boldsymbol{C}\boldsymbol{C}^T \qquad (7)$$

where $\boldsymbol{C}$ is an $B \times B$ lower triangular matrix.
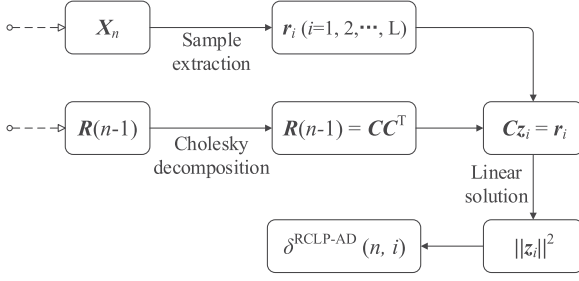
Fig. 1. Flowchart showing Cholesky decomposition-based linear system solving of RCLP-AD.

Fig. 2. Sketch map for updating $\boldsymbol{R}(n)$ in a causal linewise progressive manner, (a) $n \leq m$ and (b) $n > m$.

Accordingly, RCLP-AD could be re-expressed by

$$\delta^{\text{RCLP-AD}}(n, i) = \boldsymbol{r}_i^T (\boldsymbol{CC}^T)^{-1} \boldsymbol{r}_i$$
$$= (\boldsymbol{C}^{-1} \boldsymbol{r}_i)^T (\boldsymbol{C}^{-1} \boldsymbol{r}_i) = \|\boldsymbol{z}_i\|^2 \qquad (8)$$

where $\boldsymbol{z}_i = \boldsymbol{C}^{-1} \boldsymbol{r}_i$, a B-dimensional vector.

The main task is to determine the module, $\boldsymbol{z}_i = \boldsymbol{C}^{-1} \boldsymbol{r}_i$. It is worth noting that we do not have to invert the lower triangular matrix $\boldsymbol{C}$. Let

$$\boldsymbol{C} \boldsymbol{z}_i = \boldsymbol{r}_i. \qquad (9)$$

We can easily solve this lower triangular linear system by a forward substitution as

$$\boldsymbol{z}_i(k) = \frac{\boldsymbol{r}_i(k) - \sum_{l=1}^{k-1} \boldsymbol{C}(k, l) \boldsymbol{z}_i(l)}{\boldsymbol{C}(k, k)} \quad (k = 1, 2, \ldots, B).$$
$$(10)$$

The flowchart of this process is presented in Fig. 1.

B. Causal Progressive Line-Varying Background

Since an airborne or spaceborne imaging spectrometer usually scans across a large ground area, covering a variety of ground surfaces. To perform a better background suppression and pick up weak anomalies, it is inappropriate to use all of the data collected from the very beginning to the data line currently acquired for background suppression. In addition, some weak anomalies are usually buried in a large background but can be extracted from a small one. The term "global" for real-time anomaly detection should be redefined as an area that is in the neighborhood of the PUT (i.e., data lines marked in blue and green as shown in Fig. 2).

Suppose the number of the latest obtained data lines required for background suppression is no larger than m. If less than m lines of data have been processed, new incoming data line $\boldsymbol{X}_n$, labeled in green in Fig. 2(a), should be added to the background. This process is similar to that in [21], [25], and [23], which can be expressed as

$$\boldsymbol{R}(n) = \frac{n-1}{n} \boldsymbol{R}(n-1) + \frac{1}{n} \boldsymbol{R}(\boldsymbol{X}_n) \qquad (n \leq m) \quad (11)$$

where $\boldsymbol{R}(\boldsymbol{X}_n) = (1/L) \boldsymbol{X}_n \boldsymbol{X}_n^T$, denoting the sample correlation matrix of samples in $\boldsymbol{X}_n$.

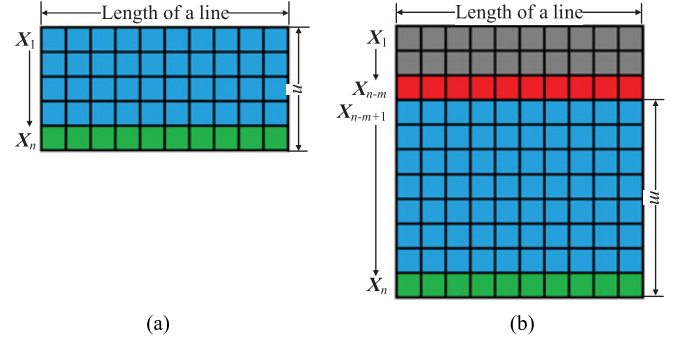Otherwise if the number of collected data lines reaches m, the process of updating $\boldsymbol{R}(n)$ involves two stages. One is to add new information $\boldsymbol{X}_n$ to the current background. The other is to discard the oldest information $\boldsymbol{X}_{n-m}$ marked in red in Fig. 2(b) from the background, derived as

$$\boldsymbol{R}(n) = \frac{1}{mL} \sum_{i=n-m+1}^{n} \boldsymbol{X}_i \boldsymbol{X}_i^T \qquad (n > m)$$

$$= \frac{1}{mL} \left(\sum_{i=n-m}^{n-1} \boldsymbol{X}_i \boldsymbol{X}_i^T - \boldsymbol{X}_{n-m} \boldsymbol{X}_{n-m}^T + \boldsymbol{X}_n \boldsymbol{X}_n^T \right)$$

$$= \boldsymbol{R}(n-1) + \frac{1}{m} [\boldsymbol{R}(\boldsymbol{X}_n) - \boldsymbol{R}(\boldsymbol{X}_{n-m})] \qquad (12)$$

where $\boldsymbol{R}(\boldsymbol{X}_{n-m}) = (1/L) \boldsymbol{X}_{n-m} \boldsymbol{X}_{n-m}^T$, denoting the sample correlation matrix of data line $\boldsymbol{X}_{n-m}$. Accordingly, $\boldsymbol{R}(n)$ could be updated line by line in a causal progressive manner, using past $\boldsymbol{R}(n-1)$, data line $\boldsymbol{X}_{n-m}$, and the current incoming data line $\boldsymbol{X}_n$.

It should be noted that the optimum m depends on the study area. For a homogenous background, which can be effectively suppressed by a small number of data lines, m should be small to identify those weak anomalies. Otherwise, more data lines are needed for better background suppression, i.e., m should be set equal to a larger number.

IV. COMPARATIVE EVALUATION: A COMPUTATIONAL LOAD PERSPECTIVE

Real-time detection is critical for the computational performance. To substantiate the high efficiency of the proposed method, detailed computational complexity analysis is essential. For comparative study on computational load, we analyzed the theoretical computational complexity of several popular linewise processing techniques that have been proposed in the literature and the one developed in this study. We would like to point out some other processing methods, such as the one proposed in [17], are not causal progressive. Instead, data lines are processed by many computing threads simultaneously in a parallel manner. However, in real-time linewise processing, we can only process one incoming data line progressively since future data lines have not been acquired.

In this study, the number of floating point operations (flops) is used for computational complexity evaluation, including the multiplication/division and addition/subtraction required for matrix operations. Prior to the computational complexity analysis, the symmetry of $\boldsymbol{R}(\boldsymbol{X}_n)$ and $\boldsymbol{R}(n)$ should be considered for further computational optimization.

A. Cholesky Decomposition-Based Linear System Solving (CDLSS)

The entire process of CDLSS-based RCLP-AD in this work can be divided into three parts:

First, Cholesky decomposition of $\boldsymbol{R}(n-1)$ as shown in (7) requires $(1/3)B^3$ flops [32].

Second, because $\boldsymbol{C}$ is a lower triangular matrix, finding the solution to the lower triangular linear system via a forward substitution as shown in (10) requires about B^2 flops for each sample. Therefore, a total of B^2L flops are required for the entire data line of L samples. $2B$ flops associated with the work to compute the norm of the vector $\boldsymbol{z}_i$ are ignored due to less significance compared to B^2 flops.

Third, updating $\boldsymbol{R}(n)$ using (11) or (12) mainly involves the calculation of the symmetric matrix, $\boldsymbol{R}(\boldsymbol{X}_n)$, with B^2L flops. It is unnecessary to recalculate $\boldsymbol{R}(\boldsymbol{X}_{n-m})$ because it has been computed in the previous process.

In conclusion, the total number of flops required for RCLP-AD based on CDLSS is

$$C_{\text{CDLSS}} = 2B^2L + (1/3)B^3. \tag{13}$$

Operations with a time complexity of $O(B^2)$, $O(L^2)$, and $O(BL)$, which are not significant compared to $O(B^3)$ and $O(B^2L)$, are not considered. The actual computing load varies slightly due to the different coding structures of the algorithms.

B. Woodbury's Identity-Based Rank-L (WIR-L) Processing

This technique [24] developed a recursive equation based on Woodbury's matrix identity to update $\boldsymbol{R}^{-1}(n)$ in a causal linewise progressive fashion, which is specified by

$$\tilde{\boldsymbol{R}}^{-1}(n) = \tilde{\boldsymbol{R}}^{-1}(n-1)$$
$$- \{\tilde{\boldsymbol{R}}^{-1}(n-1)\boldsymbol{X}_n[\boldsymbol{I} + \boldsymbol{X}_n^T\tilde{\boldsymbol{R}}^{-1}(n-1)\boldsymbol{X}_n]^{-1}$$
$$\times \boldsymbol{X}_n^T\tilde{\boldsymbol{R}}^{-1}(n-1)\} \tag{14}$$

where $\tilde{\boldsymbol{R}}(n) = nL\boldsymbol{R}(n)$, without being normalized by the number of samples. Because an expensive matrix inversion module $[\boldsymbol{I} + \boldsymbol{X}_n^T\tilde{\boldsymbol{R}}^{-1}(n-1)\boldsymbol{X}_n]^{-1}$ of rank L is required, this Woodbury's identity-based rank-L updating technique is called WIR-L processing for short.

In the light of (14), we derived a normalized recursive equation to calculate $\boldsymbol{R}^{-1}(n)$, specified by

$$\boldsymbol{R}^{-1}(n) = \frac{n}{n-1}\boldsymbol{R}^{-1}(n-1)$$
$$- \frac{n}{(n-1)^2L}\{[\boldsymbol{R}^{-1}(n-1)\boldsymbol{X}_n]$$

$$\times \left[\boldsymbol{I} + \frac{1}{(n-1)L}\boldsymbol{X}_n^T\boldsymbol{R}^{-1}(n-1)\boldsymbol{X}_n\right]^{-1}$$
$$\times [\boldsymbol{X}_n^T\boldsymbol{R}^{-1}(n-1)]\} \tag{15}$$

where the updating process is similar to that in (14). It should be noted that $\boldsymbol{R}^{-1}(n)$ is directly updated with past $\boldsymbol{R}^{-1}(n-1)$ and the incoming data line $\boldsymbol{X}_n$ using (15) without further calculation of $\boldsymbol{R}(n)$. To update $\boldsymbol{R}^{-1}(n)$ for each line, $2B^2L$ flops are needed for $\boldsymbol{R}^{-1}(n-1)\boldsymbol{X}_n$. $\boldsymbol{X}_n^T\boldsymbol{R}^{-1}(n-1)\boldsymbol{X}_n$ requires BL^2 flops considering the symmetry of the output matrix. Regarding the L-dimensional inversion module $\boldsymbol{M}_L^{-1} = [\boldsymbol{I} + (1/(n-1)L)\boldsymbol{X}_n^T\boldsymbol{R}^{-1}(n-1)\boldsymbol{X}_n]^{-1}$, the time complexity is uncertain due to various matrix inverting algorithms. As a matter of fact, $\boldsymbol{M}_L$ is also a positive definite symmetric matrix. Since $\boldsymbol{R}(n)$ is positive definite symmetric in the light of (4), $\boldsymbol{R}(n)$ can be expressed as $\boldsymbol{R}(n) = \boldsymbol{R}^{1/2}(n)\boldsymbol{R}^{1/2}(n)$. Suppose x is an $L \times 1$ nonzero vector

$$x^T\boldsymbol{M}_Lx = x^Tx + \frac{1}{(n-1)L}\left[\boldsymbol{R}^{-1/2}(n-1)\boldsymbol{X}_nx\right]^T$$
$$\times \left[\boldsymbol{R}^{-1/2}(n-1)\boldsymbol{X}_nx\right]$$
$$= \|x\|^2 + \frac{1}{(n-1)L}\|\tilde{\boldsymbol{R}}x\|^2 > 0 \tag{16}$$

where $\tilde{\boldsymbol{R}} = \boldsymbol{R}^{-1/2}(n-1)\boldsymbol{X}_n$. Thus, $\boldsymbol{M}_L$ is positive definite symmetric and could be inverted efficiently via Cholesky decomposition with $(1/3)L^3$ flops and a back substitution with $(1/3)L^3$ flops [33]. Similarly, on account of the symmetry of the output matrix, determining the product of $\boldsymbol{R}^{-1}(n-1)\boldsymbol{X}_n$, $[\boldsymbol{I} + (1/(n-1)L)\boldsymbol{X}_n^T\boldsymbol{R}^{-1}(n-1)\boldsymbol{X}_n]^{-1}$, and $\boldsymbol{X}_n^T\boldsymbol{R}^{-1}(n-1)$ needs $2BL^2 + B^2L$ flops.

For each sample, determining the product of the three items, $\boldsymbol{r}_i^T$, $\boldsymbol{R}^{-1}(n-1)$, and $\boldsymbol{r}_i$ requires $2B^2$ flops. Thus, the entire data line with L samples needs $2B^2L$ flops.

Therefore, the total number of flops required for updating $\boldsymbol{R}^{-1}(n)$ and computing the detection results for the incoming data line using (15) is

$$C_{\text{WIR}-L} = 3BL^2 + 5B^2L + (2/3)L^3 \quad (n \leq m). \tag{17}$$

The recursive equation proposed by [24] is only concerned with adding new information to the background. Thus, the computational complexity required for deleting old information is not discussed in this section.

C. Woodbury's Identity-Based Rank-B (WIR-B) Processing

In [23] and [25], a different updating formula is proposed as follows:

$$\boldsymbol{R}^{-1}(n) = \frac{n}{n-1}[\boldsymbol{R}^{-1}(n-1)$$
$$- \boldsymbol{R}_{n|n-1}(\boldsymbol{R}_{n|n-1} + \boldsymbol{I})^{-1}\boldsymbol{R}^{-1}(n-1)] \tag{18}$$

where $\boldsymbol{R}_{n|n-1} = (1/(n-1))\boldsymbol{R}^{-1}(n-1)\boldsymbol{R}(\boldsymbol{X}_n)$. In analogy with (15), the updating process of $\boldsymbol{R}(n)$ is not required. $\boldsymbol{R}^{-1}(n)$ is updated via (18) directly, involved with a rank-B matrix

inversion module, $(\boldsymbol{R}_{n|n-1} + \boldsymbol{I})^{-1}$. Similarly, we call this updating strategy as WIR-B processing.

Calculating $\boldsymbol{R}(\boldsymbol{X}_n)$ requires $B^2 L$ flops. And $2B^3$ flops are needed for $\boldsymbol{R}_{n|n-1}$. Unlike the inversion module $\boldsymbol{M}_L^{-1} = [\boldsymbol{I} + (1/(n-1)L)\boldsymbol{X}_n^T \boldsymbol{R}^{-1}(n-1)\boldsymbol{X}_n]^{-1}$ in (15), experiments on real data show that the inversion module $\boldsymbol{M}_B^{-1} = (\boldsymbol{R}_{n|n-1} + \boldsymbol{I})^{-1}$ in (18) is not positive definite, and thus cannot be inverted via Cholesky decomposition. In the light of the widely used QR decomposition-based method [10], [34]–[36], $\boldsymbol{M}_B$ could be factorized into an orthogonal matrix $\boldsymbol{Q}_M$ and an upper triangular matrix $\boldsymbol{R}_M$. Thus, determining the inversion of $\boldsymbol{M}_B$ relies on solving $\boldsymbol{R}_M^{-1}$, which is specified by

$$\boldsymbol{M}_B^{-1} = (\boldsymbol{Q}_M \boldsymbol{R}_M)^{-1} = \boldsymbol{R}_M^{-1} \boldsymbol{Q}_M^T. \tag{19}$$

QR decomposition could be completed efficiently with $2B^3$ flops using a modified Gram–Schmidt process as evidenced by [37, Section 5.2.8]. The calculation of $\boldsymbol{R}_M^{-1}$ requires $(1/3)B^3$ flops via a back substitution, while the product of $\boldsymbol{R}_M^{-1}$ and $\boldsymbol{Q}_M^T$ needs B^3 flops. Thus, inverting $\boldsymbol{M}_B$ requires a total of $(10/3)B^3$ flops. Furthermore, $3B^3$ flops are required for calculating the product of $\boldsymbol{R}_{n|n-1}$, $(\boldsymbol{R}_{n|n-1} + \boldsymbol{I})^{-1}$, and $\boldsymbol{R}^{-1}(n-1)$ considering the symmetry of the results.

For each sample, determining the product of the three items, $\boldsymbol{r}_i^T$, $\boldsymbol{R}^{-1}(n-1)$, and $\boldsymbol{r}_i$ requires $2B^2$ flops. Thus, the entire data line with L samples needs $2B^2 L$ flops.

In total, the number of flops required for updating $\boldsymbol{R}^{-1}(n)$ and solving the detection results is

$$C_{\text{WIR}-B} = 3B^2 L + (25/3)B^3 \quad (n \leq m). \tag{20}$$

Similarly, due to the lack of a process to delete old information in [23], we only analyzed the process of adding new information to the background.

D. Woodbury's Identity-Based Rank-1 (WIR-1) Processing

It is worth noting that the techniques developed in [23]–[25] involve with matrix inversion modules of rank L or B, both of which are quite expensive. In fact, we could reduce the rank-B or rank-L processing down to rank-1 processing via processing each sample in a data line one by one [15], [22]–[24], [26]–[28]. Therefore, the expensive matrix inversion process is avoided. However, the rank-1 process has to be repeated L times for the entire data line. The updating strategy for a single sample can be described as follows:

$$\boldsymbol{R}^{-1}(k) = \tilde{\boldsymbol{R}}^{-1}(k-1)$$
$$- \frac{[\tilde{\boldsymbol{R}}^{-1}(k-1)\tilde{\boldsymbol{r}}_{\boldsymbol{k}}][\tilde{\boldsymbol{r}}_{\boldsymbol{k}}^T \tilde{\boldsymbol{R}}^{-1}(k-1)]}{1 + \tilde{\boldsymbol{r}}_{\boldsymbol{k}}^T \tilde{\boldsymbol{R}}^{-1}(k-1)\tilde{\boldsymbol{r}}_{\boldsymbol{k}}} \tag{21}$$

where $\tilde{\boldsymbol{R}}^{-1}(k-1) = [(1-1/k)\boldsymbol{R}(k-1)]^{-1}$, $\tilde{\boldsymbol{r}}_{\boldsymbol{k}} = (1/\sqrt{k})\boldsymbol{r}_k$, $\boldsymbol{r}_k$ represents the kth sample of all collected data, and $\boldsymbol{R}(k)$ is the sample correlation matrix of all the samples up to the kth one.

Accordingly, a total of five modules in (21) are involved. Prior to computational complexity analysis, the symmetry of the intermediate results should be considered in order to cut down

the computing time. Namely, $\tilde{\boldsymbol{R}}^{-1}(k-1)$ requires $(1/2)B^2$ flops, the product of $\tilde{\boldsymbol{R}}^{-1}(k-1)$ and $\tilde{\boldsymbol{r}}_{\boldsymbol{k}}$ needs $2B^2$ flops. As for the numerator of the fraction in the formula, $(1/2)B^2$ flops are required, whereas the computational load of the denominator could be ignored owing to just $2B$ flops. Regarding the division of the numerator to denominator, and the subtraction of the first item to the second item in (21), both require $(1/2)B^2$ flops. Therefore, WIR-1 processing would require $4B^2$ flops for one sample, and thus $4B^2 L$ flops for one data line with L samples.

In addition, for each sample, determining the product of the three items, $\boldsymbol{r}_i^T$, $\boldsymbol{R}^{-1}(n-1)$, and $\boldsymbol{r}_i$ requires $2B^2$ flops. Thus, the entire data line with L samples needs $2B^2 L$ flops.

As a total, regardless of the process to delete old information, the computational load required to output the detection results for the entire data line is

$$C_{\text{WIR-1}} = 6B^2 L \quad (n \leq m). \tag{22}$$

E. QR Decomposition-Based Linear System Solving (QRLSS)

Linear system solving takes advantage of the linear structure of the anomaly detector [17]. As shown in (3), the intermediate vector $\boldsymbol{y}(\boldsymbol{r}_i) = \boldsymbol{R}^{-1}(n-1)\boldsymbol{r}_i$ can be efficiently defined by solving the linear system $\boldsymbol{R}(n-1)\boldsymbol{y}(\boldsymbol{r}_i) = \boldsymbol{r}_i$, instead of computing the inverse sample correlation matrix, $\boldsymbol{R}^{-1}(n)$. The detailed steps are as follows.

Step 1: Via QR decomposition using a modified Gram–Schmidt process as presented in [37, Section 5.2.8], $\boldsymbol{R}(n-1)$ could be factorized into an orthogonal matrix $\widetilde{\boldsymbol{Q}}$ and a upper triangular matrix $\widetilde{\boldsymbol{R}}$, $\boldsymbol{R}(n-1) = \widetilde{\boldsymbol{Q}}\widetilde{\boldsymbol{R}}$. Rewrite $\boldsymbol{R}(n-1)\boldsymbol{y}(\boldsymbol{r}_i) = \boldsymbol{r}_i$ as $\widetilde{\boldsymbol{Q}}\widetilde{\boldsymbol{R}}\boldsymbol{y}(\boldsymbol{r}_i) = \boldsymbol{r}_i$. This step would take $2B^3$ flops as proved in [37].

Step 2: Noting that $\widetilde{\boldsymbol{Q}}^{-1} = \widetilde{\boldsymbol{Q}}^T$, $\widetilde{\boldsymbol{R}}\boldsymbol{y}(\boldsymbol{r}_i) = \widetilde{\boldsymbol{Q}}^T \boldsymbol{r}_i$. Computing the product of $\widetilde{\boldsymbol{Q}}^T$ and $\boldsymbol{r}_i$ requires $2B^2$ flops. And the backward substitution for solving the linear system requires about B^2 flops. So this step involves approximately $3B^2$ for one sample, and thus $3B^2 L$ for one data line.

Step 3: The output filter for each sample in (3), $\delta^{\text{CLP}-\text{AD}}(n, i) = \boldsymbol{r}_i^T \boldsymbol{y}(\boldsymbol{r}_i)$, needs about $2B$ flops. The computing time associated with this step is far less significant in contrast to the work required by Steps 1 and 2.

It is worth noting that we should update $\boldsymbol{R}(n-1)$ to $\boldsymbol{R}(n)$ using (11) or (12) for the next data line. This step would require $B^2 L$ flops. To sum up, the total computational load for the entire data line is

$$C_{\text{QRLSS}} = 2B^3 + 4B^2 L. \tag{23}$$

F. Eigen Decomposition-Based Processing (ED)

The Naval Research Laboratory developed an eigen decomposition-based method to accelerate the calculation of inverse sample correlation matrix $\boldsymbol{R}(n)$ [38], [39]. Main processes are as following.

Step 1: Eigen Decomposition of $\boldsymbol{R}(n)$ via Householder reduction and QL algorithm with implicit shifts as described in

TABLE I
COMPUTATIONAL LOAD FOR DIFFERENT PROCESSES

Processing Techniques	Computational Load (flops)
CDLSS	$2B^2L + (1/3)B^3$
WIR-L	$5B^2L + 3BL^2 + (2/3)L^3$
WIR-B	$3B^2L + (25/3)B^3$
WIR-1	$6B^2L$
QRLSS	$4B^2L + 2B^3$
ED	$3B^2L + (13/3)B^3$

TABLE II
COMPUTATIONAL COMPLEXITY FOR PUSHBROOM IMAGING SPECTROMETERS

Sensor	Parameters		No. of Flops ($\times 10^7$)					
	B	L	CDLSS	WIR-L	WIR-B	WIR-1	QRLSS	ED
Hyperion	220	250	**2.8**	11.2	12.5	7.3	6.9	8.2
FTHSI	256	1024	**14.0**	185.7	34.1	40.3	30.2	27.4
HYDICE	210	320	**3.1**	15.7	12.0	8.5	7.5	8.2
SpecTIR	360	320	**9.8**	34.0	51.3	24.9	25.9	32.7
ROSIS	115	512	**1.4**	21.4	3.3	4.1	3.0	2.7
HySpex-V	108	104	**0.3**	1.0	1.4	0.7	0.7	0.9
HySpex-S	288	384	**7.2**	32.4	29.5	19.1	17.5	19.9
AISA-F	620	384	**37.5**	105.0	242.9	88.6	106.7	147.6
AISA-I	1000	384	**110.1**	240.0	948.5	230.4	353.6	548.5
CASI-2	288	101	**2.5**	5.1	22.4	5.0	8.1	12.9
PHI-1	244	376	**5.0**	25.1	18.8	13.4	11.9	13.0
PHI-2	247	652	**8.5**	69.9	24.5	23.9	18.9	18.5

[32, Chapter 11]

$$\boldsymbol{R}^{-1}(n) = \boldsymbol{Q}\boldsymbol{\Lambda}^{-1}\boldsymbol{Q}^T \qquad (24)$$

where $\boldsymbol{\Lambda}$ is a diagonal matrix with diagonal entries equal to the eigenvalues of $\boldsymbol{R}(n)$, and $\boldsymbol{Q}$ is an orthogonal matrix formed by eigenvectors corresponding to the eigenvalues in $\boldsymbol{\Lambda}$. The eigen decomposition requires $13/3B^3$ flops.

Step 2: Find the principal components $\boldsymbol{x}_i$ of each sample $\boldsymbol{r}_i$ in data line $\boldsymbol{X}_n$ by $\boldsymbol{x}_i = \boldsymbol{Q}^T\boldsymbol{r}_i$, which requires $2B^2$ flops for each sample. Therefore, a total of $2B^2L$ flops are involved for the entire data line.

Step 3: Since $\delta^{\text{CLP-AD}}(n, i) = \boldsymbol{x}_i^T\boldsymbol{\Lambda}^{-1}\boldsymbol{x}_i$, the output $\delta^{\text{CLP-AD}}(n, i)$ for sample $\boldsymbol{r}_i$ can be determined using the formula

$$\delta^{\text{CLP-AD}}(n, i) = \sum_{k=1}^{B} \frac{\boldsymbol{x}_i(k)^2}{\boldsymbol{\Lambda}(k, k)} \qquad (25)$$

where $\boldsymbol{x}_i(k)$ is the ith element of $\boldsymbol{x}_i$, and $\boldsymbol{\Lambda}(k, k)$ is the kth diagonal element of $\boldsymbol{\Lambda}$. The computational load for this step could be ignored due to rather few computing operations.

The updating process of $\boldsymbol{R}(n)$ requires B^2L flops. Therefore, eigen decomposition-based processing for the entire data line requires the computational complexity of

$$C_{\text{ED}} = (13/3)B^3 + 3B^2L. \qquad (26)$$

G. Computational Load Comparison

The computational performance of a detector depends mainly on the floating point operations it requires. Table I lists the computational load for each processing technique discussed above.

Accordingly, it is certain that CDLSS requires less time complexity than WIR-B, QRLSS, or ED does because

$$C_{\text{CDLSS}} - C_{\text{WIR-B}} = -[B^2L + (24/3)B^3] < 0$$

$$C_{\text{CDLSS}} - C_{\text{QRLSS}} = -[2B^2L + (5/3)B^3] < 0$$

$$C_{\text{CDLSS}} - C_{\text{ED}} = -4B^3 < 0. \qquad (27)$$

However, to determine whether CDLSS is less expensive than WIR-L or WIR-1, we need specific values for the number of spectral bands B and pixels L in a data line determined by imaging spectrometers.

Table II shows the technical parameters B and L for the current widely used pushbroom imaging spectrometers, along with the specific number of flops required for each of these pushbroom sensors based on various processing strategies.

As shown by the quantitative evidence in Table II (the minimum number in bold), a CDLSS-based RCLP-AD is much less expensive than do the other four strategies for the pushbroom sensors listed. In other words, RCLP-AD by virtue of CDLSS could work with high efficiency for the current pushbroom imaging spectrometers.

V. EXPERIMENTAL RESULTS AND DISCUSSION

To demonstrate that the proposed strategy CDLSS improves the efficiency of real-time CLP-AD, RCLP-AD, we used two real hyperspectral datasets collected by pushbroom imaging spectrometers for our experiments. The computer environment used for these experiments was a 64-bit windows 10 operating system with an AMD Athlon(tm) X4 750 Quad Core Processor, CPU of 3.4 GHz and RAM of 4 GB. Visual Studio 2010, with programming language C++, was used for the experimental simulation.

A. Field Imaging Spectrometer System (FISS) Data

The pushbroom FISS [40], developed by the Institute of Remote Sensing and Digital Earth under the Chinese Academy of Sciences in 2008, was used for line-by-line data collection. Based on a 464×344 charge-coupled device chip, the incoming radiation from a line of observed targets is measured spatially in a line of 464 continuous pixels and spectrally in 344 contiguous bands. The spatial resolution in terms of instantaneous field of view is about 1 mrad and the spectral resolution is better than 5 nm. Inspired by the idea of designing synthetic data in [41, Chapter 4], a real hyperspectral dataset collected by FISS was used for experiments to demonstrate the detection performance of the CDLSS-based RCLP-AD. Among the 344 bands of the images, bands 1–34 and 325–334 were removed due to a low signal-to-noise ratio. The remaining 290 bands were kept as the dataset. Fig. 3(a) shows the image scene to be studied, with 448 lines of data. A total of 18 artificial targets, with a size of 1 cm $\times$ 1 cm, 0.5 cm $\times$ 0.5 cm, and 0.2 cm $\times$ 0.2 cm were placed
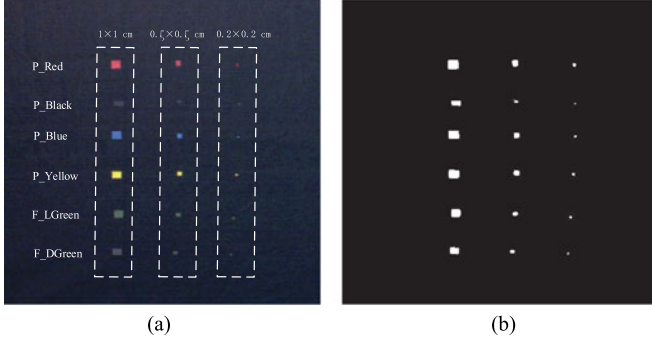
Fig. 3. (a) FISS data, with (b) ground truth.

on a large piece of black fabric as the background. There were paper targets painted in red (P_Red) in row 1, black (P_Black) in row 2, blue (P_Blue) in row 3, yellow (P_Yellow) in row 4 and fabric targets painted in light green (F_LGreen) in row 5, and dark green (F_DGreen) in row 6. The ground truth map for those targets is shown in Fig. 3(b). It should be noted that the material used for making the fabric targets was different from that used for the background.

1) Detection Results: At first, in this experiment, we set $m = +\infty$ for CDLSS in order to compare its computational performance with that of WIR-L, WIR-B, WIR-1, QRLSS, and ED. That $m = +\infty$ indicates only the addition of new information $\boldsymbol{X}_n$ is required regardless of the deletion of old information $\boldsymbol{X}_{n-m}$. The linewise progressive detection map produced by RCLP-AD based on CDLSS with $m = +\infty$ is shown in Fig. 4. The results showed that RCLP-AD runs line by line progressively. The detection maps associated with other processing strategies (i.e., WIR-R, WIR-B, WIR-1, QRLSS, and ED) are not presented because they are the same with that in Fig. 4.

Second, we set m equal to 10, 30, 100, 200, and 300, respectively, for RCLP-AD, to evaluate the detection accuracy. Namely, the process to delete the old information, $\boldsymbol{X}_{n-m}$, is considered. And thus the global background of size m slides along with the incoming data. The linewise progressive detection maps are quite similar to that shown in Fig. 4, and thus we only present the final detection maps associated with each processing techniques. To further evaluate the performance, we conducted the experiment using Global-AD. It should be noted that Global-AD is implemented in one-shot operation, with the background statistics estimated using all of the samples in a global way, not in a causal progressive manner. Thus, the background statistics used for every pixel is identical. Fig. 5 shows the final detection maps produced by Global-AD and CDLSS-based RCLP-AD with line-varying background of various size m.

The results are presented in db space for better visual inspection, where x in db is defined as $x := 20\log_{10} x$. In this experiment, the number of data lines required to calculate the initial sample correlation matrix $\boldsymbol{R}(n)$ should be greater than $\text{Int}(290/464) + 1 = 1$. However, only one line of data for the initial $\boldsymbol{R}(n)$ calculation would result in a bad detection performance for the first several data lines. Therefore, we used the first ten lines of data to calculate the initial $\boldsymbol{R}(n)$ in a global

way. The small constant λ added to the diagonal entries of $\boldsymbol{R}(n)$ was set equal to 0.000001. Subsequently, every time a new line of data are acquired, RCLP-AD could produce the output efficiently with constant computational complexity. As demonstrated in Fig. 4, the anomalies are highlighted when they occur, whereas the background is suddenly suppressed. The detection maps clearly show that the output of RCLP-AD has various intensities for different anomalies. Specifically, the intensities of anomalies in row 2 (i.e., paper anomalies painted in black) are very weak compared to those of others. By calculating the correlation coefficients between different anomalies and the background, we found that the paper anomalies painted in black are spectrally strongly correlated with the background as proved in Table III.

Through visual inspection, Global-AD almost fails to detect the paper anomalies painted in black in row 2. The main part of these black paper anomalies is buried in the global background. In contrast, these weak anomalies are better picked up by RCLP-AD as illustrated in Fig. 5(b)–(g). The results clearly demonstrate that RCLP-AD outperforms Global-AD in picking up weak anomalies. In the process of RCLP-AD, line-varying progressive background suppression is provided, making it easier for visual image interpretation, particularly when finding weak anomalies that could be overwhelmed by strong anomalies located in data lines collected later.

2) Accuracy of Detection: For a quantitative analysis of detection accuracy using the ground truth data in Fig. 3(b), we performed a three-dimensional (3-D) receiver operating characteristic (ROC) analysis as described in [15], [21], [29], [41], and [42]. Suppose $\delta^{\text{AD}}(\boldsymbol{r}_i)$ is the detection abundance fraction corresponding to the sample $\boldsymbol{r}_i$, outputted by the anomaly detector (AD). It can be normalized by

$$\delta^{\text{AD}}_{\text{normalized}}(\boldsymbol{r}_i) = \frac{\delta^{\text{AD}}(\boldsymbol{r}_i) - \min_r \delta^{\text{AD}}(\boldsymbol{r}_i)}{\max_r \delta^{\text{AD}}(\boldsymbol{r}_i) - \min_r \delta^{\text{AD}}(\boldsymbol{r}_i)} \qquad (28)$$

where $\delta^{\text{AD}}_{\text{normalized}}(\boldsymbol{r}_i)$ is the normalized abundance fraction which can be interpreted as the probability of the sample $\boldsymbol{r}_i$ to be anomalous. In the light of (28), $\delta^{\text{AD}}_{\text{normalized}}(\boldsymbol{r}_i)$ could be used to make a single decision by thresholding its output abundance fraction via a particular threshold τ, which is specified by

$$\delta^{\text{AD}}_{\tau}(\boldsymbol{r}_i) = \begin{cases} 1, & \text{if } \delta^{\text{AD}}_{\text{normalized}}(\boldsymbol{r}_i) > \tau \\ 0, & \text{otherwise} \end{cases}. \qquad (29)$$

The anomaly detector $\delta^{\text{AD}}_{\tau}(\boldsymbol{r}_i)$ in (29) is actually a new type of Neyman Pearon detector for a binary hypothesis decision problem (i.e., if $\delta^{\text{AD}}_{\text{normalized}}(\boldsymbol{r}_i)$ is greater than the given threshold τ, $\boldsymbol{r}_i$ is determined as an anomaly; otherwise, it is considered part of the background). We can obtain a parametric group of $\delta^{\text{AD}}_{\tau}(\boldsymbol{r}_i)$ through varying τ from 0 to 1. For each $\delta^{\text{AD}}_{\tau}(\boldsymbol{r}_i)$ parameterized by a specific τ, its corresponding detection probability P_D and false alarm probability P_F are produced. In conclusion, by means of $\delta^{\text{AD}}_{\tau}(\boldsymbol{r}_i)$ with varying τ, we have a number of groups of (τ, P_D, P_F) for generating a 3-D ROC curve. Furthermore, using the 3-D ROC curve, we can produce three two-dimensional

Fig. 4.   Causal progressive detection map for FISS data using CDLSS-based RCLP-AD with the sliding global background of size $m = \infty$.
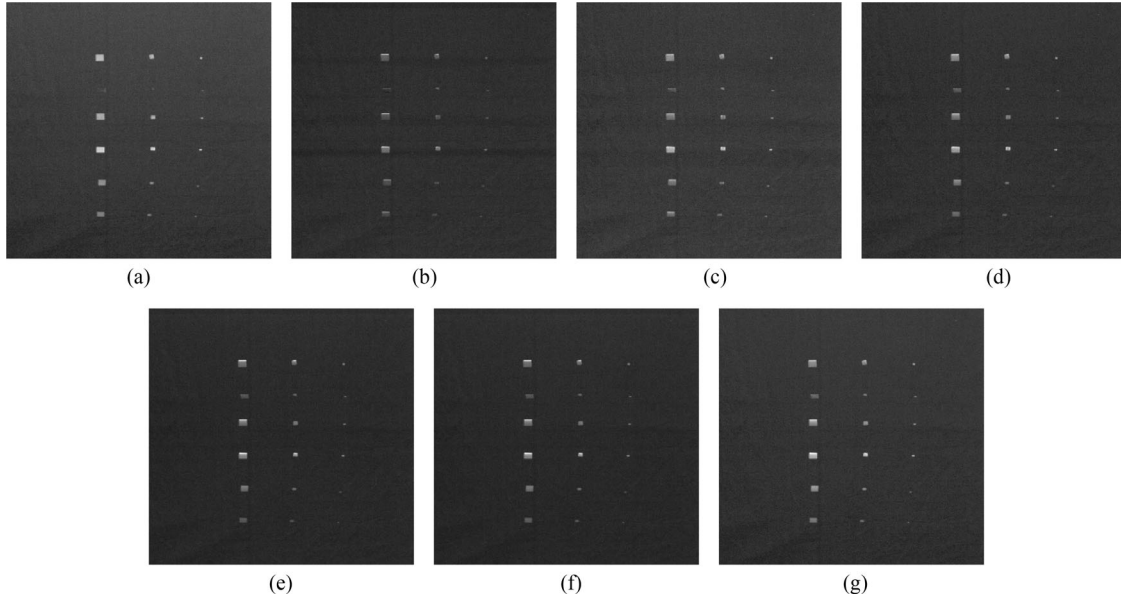

Fig. 5.   Final detection maps produced by (a) Global-AD, and CDLSS-based RCLP-AD with the sliding global background of different size m: (b) $m = 10$, (c) $m = 30$, (d) $m = 100$, (e) $m = 200$, (f) $m = 300$, (g) $m = +\infty$.

TABLE III
SPECTRAL SIMILARITY BETWEEN DIFFERENT TARGETS AND THE BACKGROUND

Anomaly	P_Red	P_Black	P_Blue	P_Yellow	F_LGreen	F_DGreen
Corr	0.8978	0.9619	0.8731	0.5429	0.9043	0.9189

(2-D) ROC curves, i.e., the 2-D ROC curve of (P_F, P_D), 2-D ROC curve of (τ, P_D), and 2-D ROC curve of (τ, P_F).

The use of 3-D ROC curves provide us with the individual progressive changes in P_D and P_F as τ varies from 0 to 1, whereas the traditional 2-D ROC curve of (P_F, P_D) cannot provide such information. In addition, anomaly detection in real-world applications is an unsupervised process with no prior knowledge of ground truth data. Thus, finding anomalies in the detection map mainly depends on visual inspection, on which background suppression has a great impact [15]. In this case, the 2-D ROC curve of (τ, P_F) provided by the 3-D ROC analysis reveals the process to suppress the background progressively. Furthermore, 3-D ROC analysis could determine the relationship between P_D and τ, and evaluate how the detection power of an anomaly detector varies with hard decisions parameterized by τ, rather than P_F.

Fig. 6 shows the ROC curves produced by Global-AD and RCLP-AD based on CDLSS with different m. To further analyze the accuracy of detection quantitatively, we calculated the area under the curve (AUC) for each 2-D ROC curve shown in Fig. 6(b)–(d) produced by Global-AD and RCLP-AD, and denoted by $A(P_F, P_D)$ for the 2-D ROC curve of (P_F, P_D), $A(\tau, P_D)$ for the 2-D ROC curve of (τ, P_D), and $A(\tau, P_F)$ for the 2-D ROC curve of (τ, P_F). The results are tabulated in Table IV, where the best results of the real-time linewise progressive anomaly detectors (i.e., RCLP-AD using CDLSS with various m) are in bold. The one-shot global detector Global-AD is also shown for comparison. It should be noted that the higher the values of $A(P_F, P_D)$ and $A(\tau, P_D)$, or the lower the value of $A(\tau, P_F)$, the better the detector performs.

As demonstrated in Fig. 6 and Table IV, we made the following conclusions.

1) RCLP-AD based on CDLSS with a small sliding global background (e.g., $m = 30$) would produce a high $A(P_F, P_D)$, which indicates a better detection power. RCLP-AD outperforms the one-shot global Global-AD. However, it does not indicate that the smaller the m, the higher the $A(P_F, P_D)$, as shown in Table IV. RCLP-
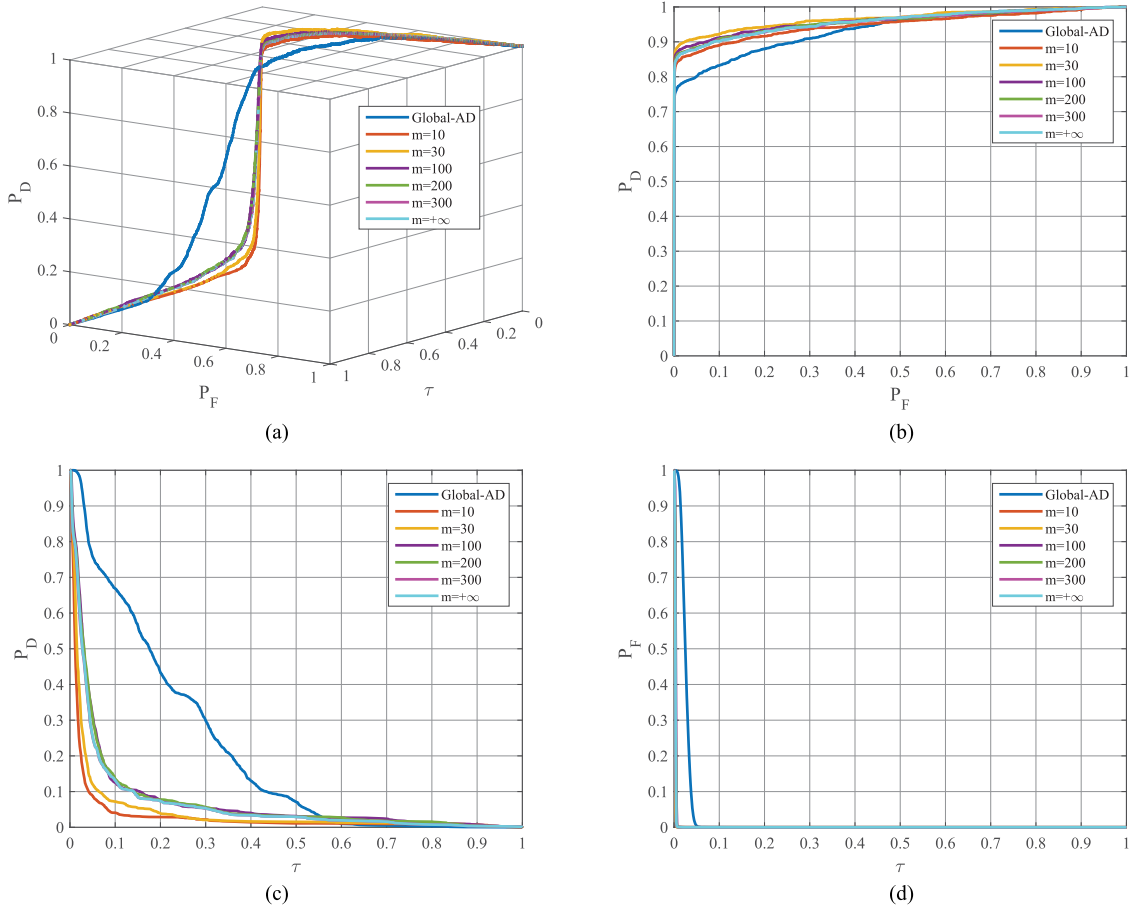
Fig. 6. Three-dimensional ROC curve and its three corresponding 2-D ROC curves for FISS data produced by Global-AD and CDLSS-based RCLP-AD with the sliding global background of different size m: (a) 3-D ROC curve of (τ, P_F, P_D), and 2-D ROC curve of (b) (P_F, P_D), (c) (τ, P_D), (d) (τ, P_F).

TABLE IV
VALUES OF THE AUC FOR EACH 2-D ROC CURVE PRODUCED BY GLOBAL-AD AND RCLP-AD BASED ON CDLSS

Algorithms	Global-AD	CDLSS-based RCLP-AD					
		$m = 10$	$m = 30$	$m = 100$	$m = 200$	$m = 300$	$m = +\infty$
$A(P_F, P_D)$	0.9361	0.9492	**0.9666**	0.9603	0.9583	0.9577	0.9584
$A(\tau, P_D)$	0.2125	0.0319	0.0420	**0.0730**	0.0716	0.0655	0.0659
$A(\tau, P_F)$	0.0256	0.0027	**0.0026**	0.0038	0.0032	0.0028	0.0030

AD based on CDLSS with $m = 10$ produced the lowest $A(P_F, P_D)$ and $A(\tau, P_D)$, indicating that the detection performance is reduced when there are too few data lines for global background estimation.

2) $A(P_F, P_D)$ is not positively correlated with $A(\tau, P_D)$. For example, Global-AD produced the lowest $A(P_F, P_D)$, but its corresponding $A(\tau, P_D)$ was much higher than that produced by RCLP-AD.

3) All causal linewise progressive anomaly detectors performed better background suppression than Global-AD does, which is implied by the smaller $A(\tau, P_F)$ of RCLP-AD.

Because the detection process is conducted progressively line by line in real time, a series of progressive continuous detection maps are produced in the process as illustrated in Fig. 4. We used $A(P_F, P_D)$ for each map to denote the detection accuracy with respect to the data line currently being processed. Fig. 7 displays the curves, with AUC values in the y-axis versus the number of data lines in the x-axis. As illustrated, the AUC values associated with CDLSS ($m=$ 30, 100, and 200) are slightly greater than that corresponding to CDLSS with $m = +\infty$. As m becomes larger, the AUC curves converge to that produced by CDLSS with $m = +\infty$. This indicates a better detection power of CDLSS-based RCLP-AD using data lines recently acquired for background estimation. However, when m is too small (i.e., too few data lines used for background estimation) the detection performance markedly degrades, as illustrated by the AUC curve corresponding to CDLSS with $m = 10$.
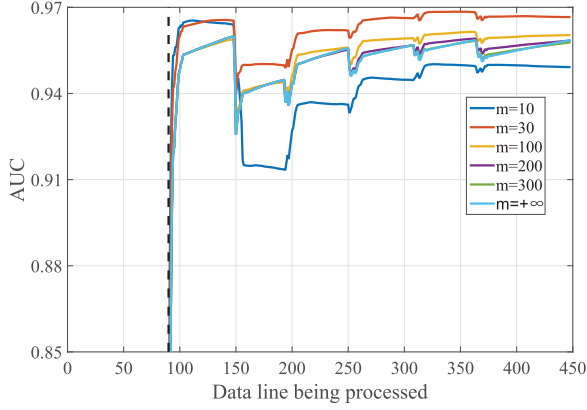
Fig. 7.   AUC curves for FISS data using CDLSS-based RCLP-AD with the sliding global background of different size m.
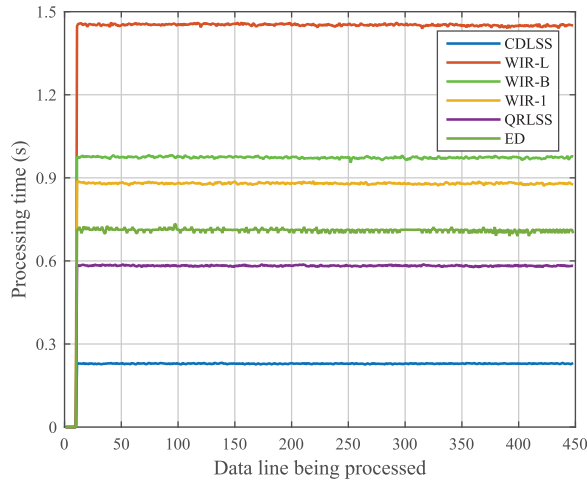

Fig. 8.   Detection time required per line for FISS data using RCLP-AD based on CDLSS, WIR-L, WIR-B, WIR-1, QRLSS, and ED.

TABLE V

AVERAGE COMPUTING TIME IN SECONDS REQUIRED TO DETECT EACH DATA LINE AND THE TOTAL TIME COST FOR PROCESSING THE ENTIRE DATASET USING RCLP-AD BASED ON CDLSS, WIR-L, WIR-B, WIR-1, QRLSS, AND ED

Algorithms	CDLSS	WIR-L	WIR-B	WIR-1	QRLSS	ED
Per line (s)	**0.229**	1.452	0.974	0.880	0.583	0.711
Total (s)	**100.2**	636.1	426.4	385.4	255.2	311.3

Bold values show that CDLSS based RCLP-AD has the best computational performance compared with other methods in terms of the real computing time per line and the total computing time for the entire dataset.

3) Computational Performance: Finally, one of the most important issues in real-time detection is the computational performance when processing each incoming line of data. In this work, we mainly focus on the computing time for each entire data line, a key indicator of the computational performance of real-time linewise progressive processing. As a matter of fact, for an onboard linewise progressive detector, it is a real-time one only if it can produce the output for the entire incoming data line, instead of just one pixel, immediately before the next one comes in.
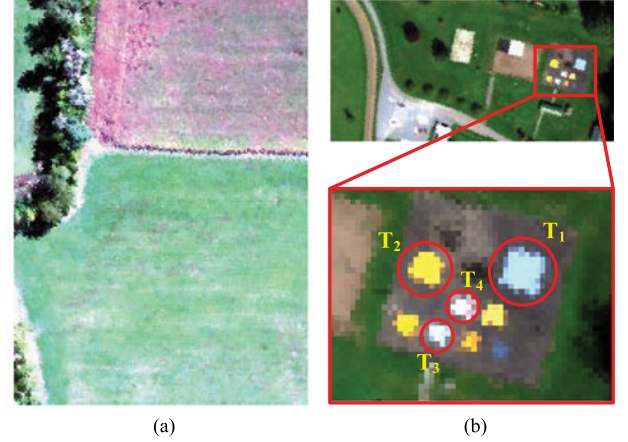

(a)   (b)

① 100% signal

② 50% signal + 50% background

③ 25% signal + 75% background

④ 12.5% signal + 87.5% background

(c)

Fig. 9.   (a) SpecTIR Avon dataset for anomaly detection, (b) real targets for implantation, (c) ground truth map for the 32 anomalies.

TABLE VI

GENERAL ACQUISITION PARAMETERS FOR THE SPECTIR AVON DATASET

Sensor System	Prospectir-VS
Spectral Range	400–2450 nm
Spectral Resolution	5 nm
Number of Spectral Bands	360
Ground Spatial Distance (GSD)	1 m

For FISS data with $B = 290$ and $L = 464$, we can determine the theoretical time complexity of CDLSS, WIR-L, WIR-B, WIR-1, and QRLSS in terms of the number of flops. $C_{\mathrm{CDLSS}} = 0.8 \times 10^8$, $C_{\mathrm{WIR\text{-}L}} = 4.5 \times 10^8$, $C_{\mathrm{WIR\text{-}B}} = 3.2 \times 10^8$, $C_{\mathrm{WIR\text{-}1}} = 2.3 \times 10^8$, $C_{\mathrm{QRLSS}} = 2.0 \times 10^8$, and $C_{\mathrm{ED}} = 2.2 \times 10^8$. This indicates that CDLSS involves much fewer flops than do the other four processing strategies. Fig. 8 shows the time consumed when processing each line of data in this experiment, where the y-axis is the processing time in seconds and the x-axis is the value of n, corresponding to the data line $\boldsymbol{X}_n$ currently being processed.

To further quantitatively evaluate the computational performance, Table V shows the average computing time required to process each data line and the entire dataset, regardless of the processing time required to detect the first ten data lines. It is clear that CDLSS-based RCLP-AD would require the least time
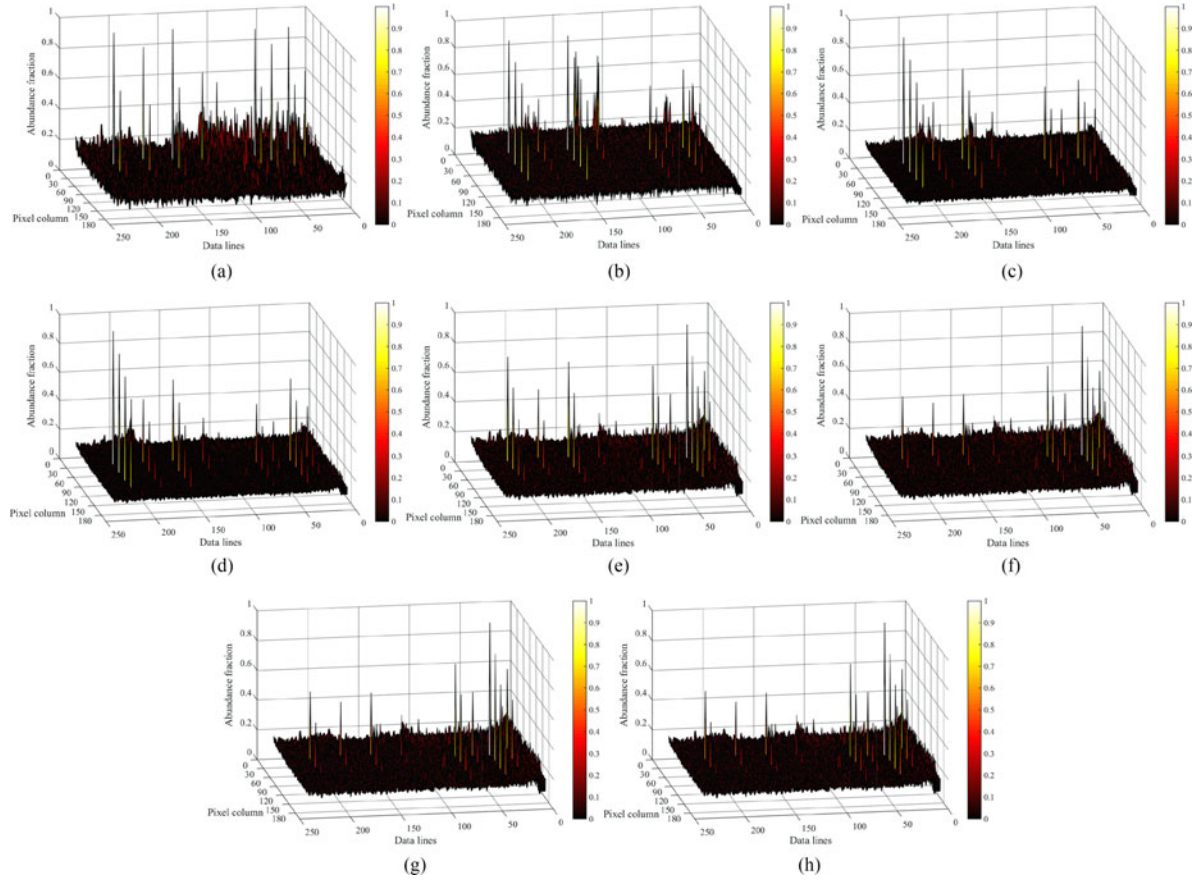
Fig. 10. Three-dimensional final detection maps produced by (a) Global-AD, and CDLSS-based RCLP-AD with (b) $m = 5$, (c) $m = 15$, (d) $m = 50$, (e) $m = 100$, (f) $m = 150$, (g) $m = 200$, (h) $m = +\infty$.

to produce the output for each data line. It should be noted that although CDLSS with $m = 10, 30, 100, 200$, and 300 involves the deletion of old information, the time consumed is almost unchanged compared to CDLSS with $m = +\infty$. More importantly, the time required to process each line of data remains almost constant, which is critical in real-time causal linewise progressive detection.

B. SpecTIR Data

SpecTIR data are a set of hyperspectral imagery data acquired line by line by the pushbroom Prospectir-VS sensor on September 9, 2012. The dataset contains several flight lines over different areas with different ground spatial resolution (GSD). In this study, we selected a subset of 234×168 pixels from the data collected over the Avon site as the study area shown in Fig. 9(a).

The main ground features in this area are grass, trees, shadows, and soil. The technical parameters regarding the SpecTIR Avon dataset are tabulated in Table VI.

Fig. 9(b) shows us an image scene of another area close to the one we subsetted in Fig. 9(a), where four real targets, denoted as T_1, T_2, T_3, and T_4 in the red rectangle, are located. Similar to the simulated data sets designed in [41, Chapter 4], we extracted four pure pixels from T_1, T_2, T_3, and T_4, respectively, and implanted them into the subsetted data in Fig. 9(a), with different abundance fractions to simulate subpixel anomalies. As shown in Fig. 9(c), each anomaly was formed by one pixel

and a total of 32 anomalies were implanted in the subsetted image scene, with abundance fractions of 100%, 50%, 25%, and 12.5% mixed with the corresponding background pixel vectors.

1) Detection Results: In this experiment, the number of data lines to ensure the nonsingularity of $\boldsymbol{R}(n)$ should be greater than $\mathrm{Int}(360/168) + 1 = 3$. As with the FISS experiment, we took the first 5 lines of data to estimate the initial global background and added the small constant $\lambda = 0.000001$ to the diagonal entries of $\boldsymbol{R}(n)$. Fig. 10 shows the 3-D plots of the detection results in db space after normalization, which were produced by Global-AD and RCLP-AD based on CDLSS, WIR-L, WIR-B, WIR-1, QRLSS, and ED. The results show that low purity (12.5%) subpixel anomalies are rarely picked up by Global-AD, as shown in Fig. 10. Through visual inspection, we found that RCLP-AD performs much better background suppression than the one-shot Global-AD. It is worth noting that, as m increases, anomalies located in data lines acquired later were rarely detected. However, the same anomalies in data lines collected earlier are better brought out from the background. The line-varying background proposed in this study enables the identification of those weak anomalies with a low purity of 12.5% and 25%, as demonstrated in Fig. 10(c) and (d), produced by CDLSS-based RCLP-AD with $m = 15$ and 50. Since the detection performance is well demonstrated by 3-D representations of the final detection maps as shown in Fig. 10, quantitative analysis is not conducted in this part.
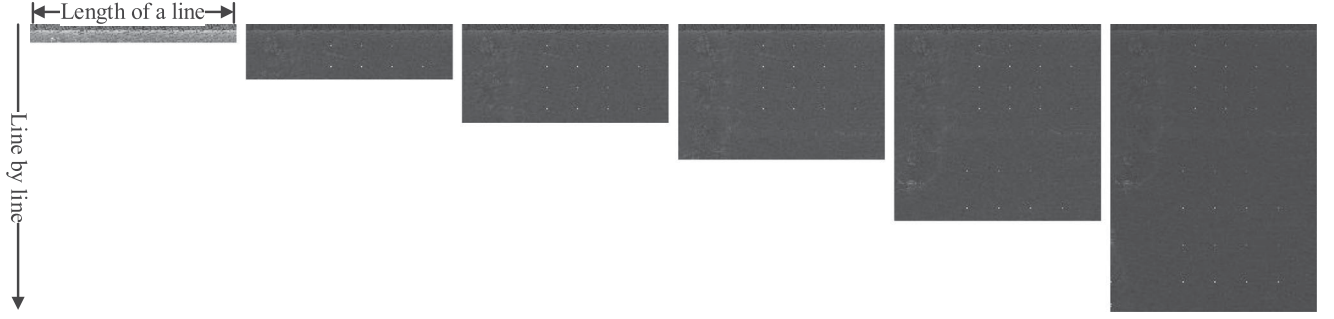
Fig. 11. Causal progressive detection map for SpecTIR data using CDLSS-based RCLP-AD with the sliding global background of size $m = 15$.
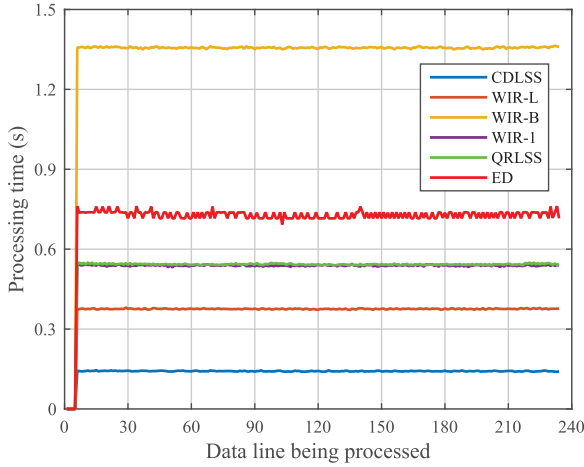

Fig. 12. Detection time required per line for SpecTIR Avon data using CLP-AD based on CDLSS, WIR-L, WIR-B, WIR-1, QRLSS, and ED.

The real-time CLP-AD process is shown in Fig. 11, where the line-varying background is suppressed by the latest 15 data lines. Before the anomalies in row 1 are revealed, the detection maps show random gray level intensities for each pixel. Then, the background is immediately suppressed as the first anomaly is revealed. When subsequent data lines come in, the background is suppressed to a different extent as shown in Fig. 11, which assist with a visual inspection to identify anomalies in the data line currently being processed.

2) Computational Performance: With regard to the issue of computational performance for SpecTIR Avon data with $B = 360$ and $L = 169$, we obtained $C_{\text{CDLSS}} = 0.6 \times 10^8, C_{\text{WIR-L}} = 1.4 \times 10^8, C_{\text{WIR-B}} = 4.5 \times 10^8, C_{\text{WIR-1}} = 1.3 \times 10^8, C_{\text{QRLSS}} = 1.8 \times 10^8$, and $C_{\text{ED}} = 2.7 \times 10^8$. This indicates the computational advantage of CDLSS. The real computing time for each data line is shown in Fig. 12. It is worth noting that the theoretical elementary floating point operations required for WIR-1 is close to that for WIR-L, but the real computing time associated with the former is more than that with the latter. This is mainly because the WIR-1 strategy processes the data pixel by pixel in a data line, too much operation expenses are required, including data flowing, memory allocating and releasing, sample vector extraction in one line of data. This would, in turn, increase the computational load for processing the entire data line. For further quantitative analysis, the average time consumed on each data line and the overall processing time for

the entire dataset, except the first five data lines, are tabulated in Table VII.

Comparing the computational performance of RCLP-AD in this experiment and the performance in the FISS experiment, we found that WIR-L is more efficient than WIR-B for the SpecTIR Avon dataset; however, this is reversed in the FISS experiment, where WIR-B is less expensive than WIR-L. The difference is mainly caused by the different technical parameters of the tested datasets (i.e., the number of spectral bands B, and the number of pixels in a data line L). However, it is clear that CDLSS-based RCLP-AD requires the minimum time to process each line of data in both experiments. Because real-time detection makes critical demands on computational performance, CDLSS is an efficient technique for real-time CLP-AD.

VI. CONCLUSION

Real-time detection requires immediate results for decision making. The computing time for processing newly collected data must be negligible. In addition, the data acquisition fashions of imaging spectrometers should be taken into consideration. Since most of the spaceborne and airborne hyperspectral imagers acquire data line by line, this study proposed several processes for RCLP-AD. The main outcomes of this work are as follows.

1) The computational complexity for processing each incoming data line was reduced significantly. Noting that the sample correlation matrix, $\boldsymbol{R}(n)$, was symmetric positive definite. Via CDLSS, the numerically unstable process of updating $\boldsymbol{R}^{-1}(n)$ can be turned into a quite stable one. Meanwhile, the entire process is significantly accelerated and simplified, which paves a way for hardware design of parallel computing.

TABLE VII
AVERAGE COMPUTING TIME IN SECONDS REQUIRED TO DETECT EACH DATA LINE AND THE TOTAL TIME REQUIRED TO PROCESS THE ENTIRE DATASET USING RCLP-AD BASED ON CDLSS, WIR-L, WIR-B, WIR-1, QRLSS, AND ED

Algorithms	CDLSS	WIR-L	WIR-B	WIR-1	QRLSS	ED
Per line (s)	**0.142**	0.376	1.357	0.539	0.543	0.729
Total (s)	**32.5**	86.0	310.7	123.4	124.4	167.0

Bold values show that CDLSS based RCLP-AD has the best computational performance compared with other methods in terms of the real computing time per line and the total computing time for the entire dataset.

2) A quantitative and comparative computational complexity analysis was undertaken in this study, after which we could find the most efficient and numerically stable technique for a specific pushbroom imaging spectrometer. Such a comparison related to real-time causal linewise progressive progressing has not been reported in the literature. Although several similar techniques have been proposed, we are not sure which one requires the least computing time before a comparative analysis. To demonstrate the computational advantage of CDLSS, we compared CDLSS with traditional algorithms (i.e., WIR-L, WIR-B, WIR-1, QRLSS, and ED) in terms of the theoretical computational load and the real computing time for processing each data line. Considering that both the length (L) of the data line and the number (B) of spectral bands have a significant impact on time consumption, we also conducted a computational performance analysis based on the pushbroom imaging spectrometers currently in wide use. Results showed that CDLSS has the superior computational performance.

3) We used a line-varying background for real-time progressive anomaly detection. Old background information was discarded when background statistics were updated progressively. The deletion process was particularly designed from the perspective of a real onboard application. Moreover, some weak anomalies were well identified with a line-varying background. Using traditional updating strategies (i.e., all collected data lines involved in background suppression), such weak anomalies are usually buried in the progressive detection process. Additionally, it should be noted that although the operation to delete old information was conducted during the process, the computational load for processing each data line was unchanged.

4) RCLP-AD with a line-varying background provided progressive background suppression for better visual image interpretation. Because anomaly detection was mainly based on the contrast of an anomaly relative to the background, but not the gray level intensities of an anomaly, line-varying background suppression enabled the identification of weak anomalies before they were overwhelmed by strong anomalies that were detected later.

The future work following this study will be focused on hardware design of RCLP-AD, such as FPGA parallel computing. In that case, the data stream rate, including acquisition and processing, should be taken into consideration for onboard (e.g., UAV or satellite) application of RCLP-AD.

ACKNOWLEDGMENT

REFERENCES

[1] R. A. Schowengerdt, *Remote Sensing: Models and Methods for Image Processing*. New York, NY, USA: Academic, 2006.

[2] A. F. Goetz, "Three decades of hyperspectral remote sensing of the earth: A personal view," *Remote Sens. Environ.*, vol. 113, pp. S5–S16, 2009.

[3] K. L. Castro-Esau, G. A. Sánchez-Azofeifa, B. Rivard, S. J. Wright, and M. Quesada, "Variability in leaf optical properties of mesoamerican trees and the potential for species classification," *Amer. J. Botany*, vol. 93, no. 4, pp. 517–530, 2006.

[4] C.-I. Chang, *Hyperspectral Imaging: Techniques for Spectral Detection and Classification*, vol. 1. New York, NY, USA: Springer, 2003.

[5] S. Matteoli, M. Diani, and G. Corsini, "A tutorial overview of anomaly detection in hyperspectral images," *IEEE Aerosp. Electron. Syst. Mag.*, vol. 25, no. 7, pp. 5–28, Jul. 2010.

[6] N. M. Nasrabadi, "Hyperspectral target detection: An overview of current and future challenges," *IEEE Signal Process. Mag.*, vol. 31, no. 1, pp. 34–44, Jan. 2014.

[7] D. Manolakis, E. Truslow, M. Pieper, T. Cooley, and M. Brueggeman, "Detection algorithms in hyperspectral imaging systems: An overview of practical algorithms," *IEEE Signal Process. Mag.*, vol. 31, no. 1, pp. 24–33, Jan. 2014.

[8] I. S. Reed and X. Yu, "Adaptive multiple-band cfar detection of an optical pattern with unknown spectral distribution," *IEEE Trans. Acoust. Speech Signal Process.*, vol. 38, no. 10, pp. 1760–1770, Oct. 1990.

[9] C.-I. Chang and S.-S. Chiang, "Anomaly detection and classification for hyperspectral imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 40, no. 6, pp. 1314–1325, Jun. 2002.

[10] B. Du and L. Zhang, "Random-selection-based anomaly detector for hyperspectral imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 49, no. 5, pp. 1578–1589, May 2011.

[11] W. Li and Q. Du, "Collaborative representation for hyperspectral anomaly detection," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 3, pp. 1463–1474, Mar. 2015.

[12] J. A. Jablonski, T. J. Bihl, and K. W. Bauer, "Principal component reconstruction error for hyperspectral anomaly detection," *IEEE Geosci. Remote Sens. Lett.*, vol. 12, no. 8, pp. 1725–1729, Aug. 2015.

[13] H. Kwon and N. M. Nasrabadi, "Kernel rx-algorithm: A nonlinear anomaly detector for hyperspectral imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 43, no. 2, pp. 388–397, Feb. 2005.

[14] T. Wu, L. Zhang, B. Peng, H. Zhang, Z. Chen, and M. Gao, "Real-time progressive hyperspectral remote sensing detection methods for crop pest and diseases," *Proc. SPIE*, vol. 9874, 2016, Art. no. 987410.

[15] S.-Y. Chen, Y. Wang, C.-C. Wu, C. Liu, and C.-I. Chang, "Real-time causal processing of anomaly detection for hyperspectral imagery," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 50, no. 2, pp. 1511–1534, Apr. 2014.

[16] C.-I. Chang, Y. Li, M. C. Hobbs, R. C. Schultz, and W. M. Liu, "Progressive band processing of anomaly detection in hyperspectral imagery," *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.*, vol. 8, no. 7, pp. 3558–3571, Jul. 2015.

[17] J. M. Molero, E. M. Garzn, I. Garca, and A. Plaza, "Analysis and optimizations of global and local versions of the rx algorithm for anomaly detection in hyperspectral data," *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.*, vol. 6, no. 2, pp. 801–814, Apr. 2013.

[18] A. Paz and A. Plaza, "Clusters versus gpus for parallel target and anomaly detection in hyperspectral images," *EURASIP J. Adv. Signal Process.*, vol. 2010, 2010, Art. no. 915639.

[19] B. Zhang, W. Yang, L. Gao, and D. Chen, "Real-time target detection in hyperspectral images based on spatial-spectral information extraction," *EURASIP J. Adv. Signal Process.*, vol. 2012, no. 1, pp. 1–15, 2012.

[20] B. Yang, M. Yang, A. Plaza, L. Gao, and B. Zhang, "Dual-mode FPGA implementation of target and anomaly detection algorithms for real-time hyperspectral imaging," *IEEE J. Sel. Topics Appl. Earth Obs. Remote Sens.*, vol. 8, no. 6, pp. 2950–2961, Jun. 2015.

[21] C.-I. Chang, *Real-Time Progressive Hyperspectral Image Processing: Endmember Finding and Anomaly Detection*. New York, NY, USA: Springer, 2016.

[22] C. Zhao, Y. Wang, B. Qi, and J. Wang, "Global and local real-time anomaly detectors for hyperspectral remote sensing imagery," *Remote Sens.*, vol. 7, no. 4, pp. 3966–3985, 2015.

[23] C.-I. Chang, H.-C. Li, M. Song, C. Liu, and L. Zhang, "Real-time constrained energy minimization for subpixel detection," *IEEE J. Sel. Topics Appl. Earth Obs. Remote Sens.*, vol. 8, no. 6, pp. 2545–2559, Jun. 2015.

[24] Q. Du and R. Nekovei, "Fast real-time onboard processing of hyperspectral imagery for detection and classification," *J. Real-Time Image Process.*, vol. 4, no. 3, pp. 273–286, 2009.

[25] H.-C. Li and C.-I. Chang, "Real-time hyperspectral anomaly detection via band-interleaved by line," *Proc. SPIE*, vol. 9874, 2016, Art. no. 98740H.

[26] N. Acito, S. Matteoli, M. Diani, and G. Corsini, "Complexity-aware algorithm architecture for real-time enhancement of local anomalies in hyperspectral images," *J. Real-Time Image Process.*, vol. 8, no. 1, pp. 53–68, 2011.

[27] A. Rossi, N. Acito, M. Diani, and G. Corsini, "Computationally efficient strategies to perform anomaly detection in hyperspectral images," *Proc. SPIE*, vol. 8537, 2012, Art. no. 85370H.

[28] A. Rossi, N. Acito, M. Diani, and G. Corsini, "Rx architectures for real-time anomaly detection in hyperspectral images," *J. Real-Time Image Process.*, vol. 9, no. 3, pp. 503–517, 2014.

[29] C.-I. Chang, Y. Wang, and S. Y. Chen, "Anomaly detection using causal sliding windows," *IEEE J. Sel. Topics Appl. Earth Obs. Remote Sens.*, vol. 8, no. 7, pp. 3260–3270, Jul. 2015.

[30] G. Stewart, *Matrix Algorithms: Basic Decompositions*, vol. 1. Philadelphia, PA, USA: Soc. Ind. Appl. Math., 1998.

[31] M. Seeger, "Low rank updates for the cholesky decomposition," EPFL, Lausanne, Switzerland, Tech. Rep. EPFL-REPORT-161468, 2004.

[32] W. H. Press, *Numerical Recipes: The Art of Scientific Computing*, 3rd ed. Cambridge, U.K.: Cambridge Univ. Press, 2007.

[33] A. Krishnamoorthy and D. Menon, "Matrix inversion using cholesky decomposition," in *Proc. Signal Process., Algorithms Architectures Arrangements Appl.*, 2013, pp. 70–72.

[34] C.-I. Chang, H. Ren, and S. S. Chiang, "Real-time processing algorithms for target detection and classification in hyperspectral imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 39, no. 4, pp. 760–768, Apr. 2001.

[35] C.-I. Chang and M. Hsueh, "Characterization of anomaly detection in hyperspectral imagery," *Sensor Rev.*, vol. 26, no. 2, pp. 137–146, 2006.

[36] M. Hsueh, "Reconfigurable computing for algorithms in hyperspectral image processing," Ph.D. dissertation, Univ. Maryland, Baltimore, MD, USA, 2007.

[37] G. H. Golub and C. F. Van Loan, *Matrix Computations*, vol. 3. Baltimore, MD, USA: JHU Press, 2012.

[38] C. M. Stellman, G. G. Hazel, F. Bucholtz, J. V. Michalowicz, A. Stocker, and W. Schaaf, "Real-time hyperspectral detection and cuing," *Opt. Eng.*, vol. 39, no. 7, pp. 1928–1935, 2000.

[39] B. Stevenson *et al.*, "Design and performance of the civil air patrol archer hyperspectral processing system," *Proc. SPIE*, vol. 5806, pp. 731–742, 2005.

[40] Q. Tong, Y. Xue, and L. Zhang, "Progress in hyperspectral remote sensing science and technology in china over the past three decades," *IEEE J. Sel. Topics Appl. Earth Obs. Remote Sens.*, vol. 7, no. 1, pp. 70–91, Jan. 2014.

[41] C.-I. Chang, *Hyperspectral Data Processing: Algorithm Design and Analysis*. Hoboken, NJ, USA: Wiley, 2013.

[42] C.-I. Chang, "Multiparameter receiver operating characteristic analysis for signal detection and classification," *IEEE Sensors J.*, vol. 10, no. 3, pp. 423–442, Mar. 2010.
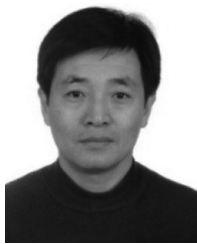
Lifu Zhang (S'04–M'05–SM'14) received the B.E. degree in photogrammetry and remote sensing from the Department of Airborne Photogrammetry and Remote Sensing, Wuhan Technical University of Surveying and Mapping (WTUSM), Wuhan, China, in 1992, the M.E. degree in photogrammetry and remote sensing from the State Key Laboratory of Information Engineering in Surveying, Mapping and Remote Sensing, WTUSM, in 2000, and the Ph.D. degree in photogrammetry and remote sensing from the State Key Laboratory of Information Engineering in Surveying, Mapping and Remote Sensing, Wuhan University, Wuhan, in 2005.

He was a Visiting Researcher in the Department of Information and Computer Sciences, Nara Women's University, Nara, Japan, from 2003 to 2004, a Postdoctoral Researcher with the Institute of Remote Sensing and Geographic Information System, School of Earth and Space Sciences, Peking University, Beijing, China, from 2005 to 2007, and an Advanced Visiting Researcher with the Earth Science and Resource Engineering, Commonwealth Scientific and Industrial Research Organization (CSIRO), Sydney, NSW, Australia, in 2011. He is currently a Fulltime Professor and the Dean of the Hyperspectral Remote Sensing Division, Institute of Remote Sensing and Digital Earth, Chinese Academy of Sciences, Beijing. His research interests include hyperspectral remote sensing, imaging spectrometer system development and its applications.

Dr. Zhang is a member of SPIE, a senior member of the Academy of Space Science of China, and is also a committeeman of Chinese National Committee of the International Society for Digital Earth (CNISDE), a Vice Chairman of Hyperspectral Earth Observation Committee (HEOC) of CNISDE, and a Standing Committeeman of the Expert Committee of China Association of Remote Sensing Applications.

Bo Peng (S'16) received the B.E. degree in remote sensing science and technology from Wuhan University, Wuhan, China, in 2014. He is currently working toward the M.S. degree in cartography and geographic information system at the University of Chinese Academy of Sciences, Beijing, China.
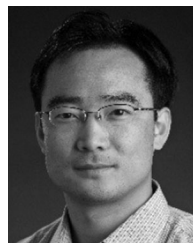
He is a Graduate Research Assistant with the Hyperspectral Remote Sensing Division, Institute of Remote Sensing and Digital Earth, Chinese Academy of Sciences, Beijing, China. His research interests include hyperspectral signal and image processing, and high-performance computing.

Feizhou Zhang received the B.E. degree in industry electric automation from the Department of Electronic and Electric Engineering, Shanxi Technology College, Xian, China, in 1986, the M.E. and Ph.D. degrees in navigation, control and guide from the School of Automatic Science, University of Aeronautics and Astronautics (BUAA), Beijing, China, in 1997 and 2001, respectively.

He is currently an Associate Professor and the Vice Dean in the School of Earth and Spatial Sciences, Peking University, Beijing. His research interests include the research in satellite navigation, Internet of Things, software receiver, ITS, spatial information integration, etc. He has published more than 100 papers among, which over 70 are indexed by SCI and EI.

Lizhe Wang received the B.E. and M.E. degrees in Electrical Engineering from Tsinghua University, Beijing, China, in 1998 and 2001 respectively, and the D.Eng. degree (Magna Cum Laude) in applied computer science from the University of Karlsruhe, Karlsruhe, Germany, in 2007.

He is a "ChuTian" Chair Professor in the School of Computer Science, China University of Geosciences (CUG), and a Professor with the Institute of Remote Sensing and Digital Earth, Chinese Academy of Sciences, Beijing. His main research interests include HPC, e-Science, and remote sensing image processing.

Prof. Wang is a Fellow of IET, and the British Computer Society. He serves as an Associate Editor of IEEE TRANSACTIONS ON PARALLEL AND DISTRIBUTED SYSTEMS, IEEE TRANSACTIONS ON CLOUD COMPUTING, and IEEE TRANSACTIONS ON SUSTAINABLE COMPUTING.

Hongming Zhang received the B.E. degree from Shandong University, Shandong, China, in 2008, and the Ph.D. degree from Huazhong University of Science and Technology, Beijing, China, in 2013, both in biomedical engineering.

He is currently an Associate Professor with the Institute of Remote Sensing and Digital Earth, Beijing. His research interests include the development of hyperspectral imaging system and food quality inspection.
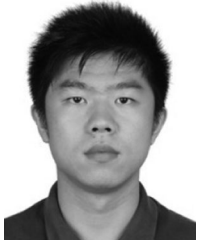
Peng Zhang was born in Shanxi, China. He is currently working toward the Ph.D. degree in cartography and geographic information system at the Institute of Remote Sensing and Digital Earth, Chinese Academy of Sciences, University of Chinese Academy of Sciences, Beijing, China.

His research interests include UAV and imaging platform development, hyperspectral image geometric rectification, and infrared oblique thermography technique.

Qingxi Tong received the B.S. degree in agrometeorology from Odessa of Hydro meteorological Institute, Odessa, Ukraine, in 1961.

He has been engaged in the study on the development and application of remote sensing since the beginning of 1970. He is currently the Chief Scientist in a cooperation project of the National Remote Sensing Center of China (NRSCC) and the Surrey Satellite Technology Limited (SSTL), Guildford, U.K., for the development of a small satellite. As a result, a high-performance small satellite named "Beijing-1" has been launched and successfully operated. He is one of the Principal Scientists in remote sensing in China. He has made outstanding contribution in the development of remote sensing technology and applications, particularly in the development of airborne remote sensing system, and in the study of remote sensing spectral properties of Earth resources and environments.

Prof. Tong received the National Prizes for Progress of Science and Technology and the Chinese Academy of Sciences (CAS) Prize for Progress of Science and Technology many times due to his achievements. He also received the Achievement for International Remote Sensing Science and Technology by the SPIE in 2002. He is an Associate Editor of *Journal of Remote Sensing*, and an Editorial Committeeman of *Chinese Journal of Space Science*. He is also active in the international cooperation. He was elected as a member of CAS and an Academician of the International Eurasian Academy of Sciences in 1997. He serves as the Chairman of the Expert Committee of NRSCC, and the Ministry of Science and Technology of China.