

Research Article

Key-Frame Detection and Super-Resolution of Hyperspectral Video via Sparse-Based Cumulative Tensor Factorization

Ruofei Zhou ¹, Gang Wang ¹, Bo Li ¹, Jinlong Wang,¹ Tianzhu Liu,^{1,2} and Chungang Liu³

¹School of Electronics and Information Engineering, Harbin Institute of Technology, Harbin 150001, China

²Heilongjiang Province Key Laboratory of Space-Air-Ground Integrated Intelligent Remote Sensing, Harbin 150001, China

³College of Career Technology, Hebei Normal University, Shijiazhuang 050000, China

Correspondence should be addressed to Gang Wang; gwang51@hit.edu.cn

Received 11 May 2020; Revised 25 May 2020; Accepted 5 June 2020; Published 14 July 2020

Guest Editor: Liangtian Wan

Copyright © 2020 Ruofei Zhou et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Thanks to the rapid development of hyperspectral sensors, hyperspectral videos (HSV) can now be collected with high temporal and spectral resolutions and utilized to handle invisible dynamic monitoring missions, such as chemical gas plume tracking. However, using such sequential large-scale data effectively is challenged, because the direct process of these data requires huge demands in terms of computational loads and memory. This paper presents a key-frame and target-detecting algorithm based on cumulative tensor CANDECOMP/PARAFAC (CP) factorization (CTCF) to select the frames where the target shows up, and a novel super-resolution (SR) method using sparse-based tensor Tucker factorization (STTF) is used to improve the spatial resolution. In the CTCF method, the HSV sequence is seen as cumulative tensors and the correlation of adjacent frames is exploited by applying CP tensor approximation. In the proposed STTF-based SR method, we consider the HSV frame as a third-order tensor; then, HSV frame super-resolution problem is transformed into estimations of the dictionaries along three dimensions and estimation of the core tensor. In order to promote sparse core tensors, a regularizer is incorporated to model the high spatial-spectral correlations. The estimations of the core tensor and the dictionaries along three dimensions are formulated as sparse-based Tucker factorizations of each HSV frame. Experimental results on real HSV data set demonstrate the superiority of the proposed CTCF and STTF algorithms over the comparative state-of-the-art target detection and SR approaches.

1. Introduction

Hyperspectral imaging has been one of the most popular research fields due to its ability of identifying the materials from very high spectral resolution and coverage. In the last decade, researchers focused on the processing and application of hyperspectral image (HSI), such as denoising [1, 2], feature extraction [3, 4], classification [5–11], detection [12–14], and super-resolution (fusion) [15–18]. In this section, researching of the latter two fields which are related to this paper will be briefly introduced.

Basically, target detection is a kind of binary classifier with the purpose of labeling every image pixel as a target or background. In HSIs, pixels with a significantly different spectral signature from their neighboring background pixels are defined as spectral anomalies. Anomaly detectors are

statistical or pattern recognition methods used to detect distinct pixels that differ from the background. It is worth mentioning that, in spectral anomaly detection approaches [19–22], such as Reed-Xiao (RX) algorithm [23], no prior information of the target spectral signature is assumed or used. However, we focus on the detection of invisible gas plumes in this paper, and the prior knowledge of the desired targets spectral characteristics is assumed to be known. In such cases, signature-based target detection algorithms are presented instead of anomaly detection. In these algorithms, the spectral characteristics of the target can be represented by a target subspace or a single target spectrum [24]. Likewise, the characteristics of background can be statistically expressed by a Gaussian distribution or a subspace defining the local or whole background statistics. As for this category, the matched subspace detector (MSD) method [25]

is one of the most typical algorithms. In the MSD, the target pixel vectors are represented by a linear combination of the target spectral signature and the background spectral signature, which stand for the subspace target spectra and the subspace background spectra, respectively. Then, the generalized likelihood ratio test (GLRT) is applied, using projection matrices associated with the background subspace and the target-and-background subspace. At last, the comparison between the output of GLRT and a preset threshold makes a final decision about whether the target is absent or present. From pixel level to subpixel level, a single pixel may contain several distinct pure materials (endmembers), also known as the mixed pixel. The presence of mixed pixels is a tough problem caused by the low spatial resolution of HSIs. Accordingly, some unmixing approaches [26–28] have been designed to compute fractional abundance of endmembers. In [29], a hyperspectral unmixing approach based on constrained matrix factorization (CMF) was proposed. Unlike conventional methods, each column vector of endmember matrix is represented as a nonnegative linear combination of pixel spectra. After endmember matrix and the corresponding fractional abundance matrix are obtained by solving optimization problems, abundance map of the target endmember shows the detection result.

As mentioned before, the HSIs often suffered from low spatial resolution. To acquire an HSI, the number of sun photons in each spectral band has to be greater than a minimum value, and the number of spectral bands is so huge in an HSI that the spatial resolution has to be sacrificed. Therefore, super-resolution (SR) techniques have aroused great interest in the last decade. Generally, the SR methods of HSI can be classified into four categories: Bayesian [30], component analysis [31], deep learning [32], and sparse representation. Due to the limited length of this paper, we focus on the introduction of sparse-based algorithms. In such HSI super-resolution schemes, images are expressed by dictionaries and corresponding sparse coefficients. On the basis of the spatial-spectral sparsity in the HSIs, the dictionaries and sparse coefficients are estimated jointly [33]. Huang et al. [34] introduced a fusion method of multispectral images (MSIs) with different spectral and spatial resolutions based on sparse matrix factorization. Akhtar et al. [35] presented an MSI-HSI fusion approach using sparse coding and Bayesian dictionary learning. Moreover, some algorithms based on matrix factorization [36–38] or unmixing [39] can also be regarded as the sparse representation schemes because the source images are decomposed into some basis and the corresponding coefficients. Yokoya et al. proposed a couple nonnegative matrix factorization (CNMF) [40] algorithm, where the unmixing techniques are employed to yield the endmember matrices and the high-resolution (HR) abundance matrices of HSI. In [41], Lanaras et al. suggested a joint scheme to solve the spectral unmixing problems. In [42], Zhang et al. fused the low-resolution (LR) HSI and HR-MSI based on the group spectral embedding and low-rank factorization.

However, the matrix factorization based schemes cannot fully exploit the spatial-spectral correlations of the HSIs. It is believed that considering HSIs as tensors is better because an HSI can be naturally expressed as a third-order tensor. In this paper, a detection algorithm based on cumulative tensor CP factorization (CTCF) is proposed. The sequential HSV data is expressed as a four-dimensional (4D) cumulative tensor; factor matrices are obtained by decomposing original 4D tensor using CP factorization. When a new frame presents and is added to the time dimension of the original tensor, this 4D cumulative tensor is updated together with the factor matrices. Consequently, a CP tensor approximation of the new frame is computed by updated factor matrices and the fitness between the new frame and the approximation is calculated. After comparing the fitness to a preset threshold, we can make the decision that whether the new frame continues to be used to update the cumulative tensor or the new frame is the key-frame where the target presents. CTCF-based method exploits not only the spatial-spectral correlations of the HSIs by applying tensor model, but also the temporal correlation between adjacent frames of the HSV.

On the other hand, tensor-based analysis has also been widely used in HSI super-resolution [43–45]. To the best of our knowledge, most of the SR algorithms enhance spatial resolution by fusing high-resolution MSI (HR-MSI) and low-resolution HSI (LR-HSI) from the same scene. Unfortunately, it is less practical in real application. In some situations, LR-HSI is the only data we have rather than both. In this paper, we suggest an SR algorithm using sparse-based tensor Tucker factorization (STTF). Inspired by the Tucker factorization and its related works, the HSV frames are represented as third-order tensors, which are approximated by the multiplication of the dictionaries along three dimensions (i.e., the dictionaries of the height mode, the width mode, and the spectral mode: they are named “three modes dictionaries” for short in the rest of this paper) and a core tensor. Then, the problem of SR is transformed into the estimations of the three modes dictionaries and estimation of the core tensor. Specifically, the spatial information is represented by the height mode dictionary and the width mode dictionary, the spectral information is represented by the spectral mode dictionary, and the correlations of the three modes dictionaries are modeled by the core tensor. HSIs are generally self-similar so that a sparse prior can be imposed on the core tensor; then, the estimations of the core tensor and three modes dictionaries are formulated as the STTF of the LR and HR HSV frames. In the iteration of STTF, core tensor and dictionaries are all updated and accurate estimates are yielded when convergence is achieved.

The remainder of this paper is organized as follows. Section 2 presents the materials and methods, including the basic notations and preliminaries of tensor and tensor factorization, the proposed CTCF approach for key-frame detection, and the proposed STTF method for key-frame super-resolution problem. In Section 3, experimental results on real HSV and the discussions are given. The paper is

summarized in Section 4 with ideas for future work along the path presented here.

2. Materials and Methods

2.1. Tensor Notations and Preliminaries

2.1.1. Tensor Notations. In this paper, vectors are denoted by boldface lowercase letters ($\mathbf{a}, \mathbf{b}, \mathbf{c}, \dots$), matrices are denoted by boldface capital letters ($\mathbf{A}, \mathbf{B}, \mathbf{C}, \dots$), and tensors are denoted by bold Euler script letters ($\mathbf{A}, \mathbf{B}, \mathbf{C}, \dots$). Generally, a tensor is a kind of multidimensional array, denoted by $\mathbf{A} \in \mathbb{R}^{I_1 \times I_2 \times \dots \times I_N}$. Here, tensor \mathbf{A} is an N th-order tensor and I_n ($1 \leq n \leq N$) is the dimension of the n th mode. Obviously, vectors are first-order tensors and matrices are second-order tensors. We use $\mathbf{A}(i_1, \dots, i_{n-1}, :, i_{n+1}, \dots, i_N)$ to denote the mode- n fiber, which are vectors yielded from tensor \mathbf{A} by changing index i_n with other indexes fixed. The mode- n unfolding matrix of tensor \mathbf{A} is generated by placing all the mode- n fibers in a matrix as columns, denoted by $\mathbf{A}_{(n)} \in \mathbb{R}^{I_n \times I_1 \times \dots \times I_{n-1} \times I_{n+1} \times \dots \times I_N}$.

An important calculation between a tensor and a matrix is the n -mode product, which is defined as

$$\mathbf{F} = \mathbf{A} \times_n \mathbf{B}, \quad (1)$$

where $\mathbf{B} \in \mathbb{R}^{I_n \times I_n}$ and $\mathbf{F} \in \mathbb{R}^{I_1 \times \dots \times I_{n-1} \times J_n \times I_{n+1} \times \dots \times I_N}$. The elements of \mathbf{A} are denoted by $a_{i_1 i_2 \dots i_N}$, so the elements of \mathbf{F} are computed by

$$f_{i_1, \dots, i_{n-1}, j_n, i_{n+1}, \dots, i_N} = \sum_{i_n} a_{i_1, \dots, i_{n-1}, i_n, i_{n+1}, \dots, i_N} b_{j_n i_n}. \quad (2)$$

Given the definition of n -mode product, we can obtain

$$\mathbf{F} = \mathbf{A} \times_n \mathbf{B} \Leftrightarrow \mathbf{F}_{(n)} = \mathbf{B} \mathbf{A}_{(n)}. \quad (3)$$

For continuous multiplication of a tensor and matrices in distinct modes, the result is not affected by the multiplication order, described by

$$\mathbf{A} \times_n \mathbf{B} \times_m \mathbf{C} = \mathbf{A} \times_m \mathbf{C} \times_n \mathbf{B} \quad (m \neq n). \quad (4)$$

If the modes are equivalent, equation (4) is transformed into

$$\mathbf{A} \times_n \mathbf{B} \times_n \mathbf{C} = \mathbf{A} \times_n (\mathbf{C} \mathbf{B}). \quad (5)$$

Suppose that $\mathbf{E}_n \in \mathbb{R}^{J_n \times I_n}$ ($1 \leq n \leq N$) is a collection of matrices; we define tensor $\mathbf{G} \in \mathbb{R}^{J_1 \times J_2 \times \dots \times J_N}$ as

$$\mathbf{G} = \mathbf{A} \times_1 \mathbf{E}_1 \times_2 \mathbf{E}_2 \cdots \times_N \mathbf{E}_N. \quad (6)$$

The matricization form of equation (6) is presented by

$$\mathbf{g} = (\mathbf{E}_N \otimes \mathbf{E}_{N-1} \otimes \dots \otimes \mathbf{E}_1) \mathbf{a}, \quad (7)$$

where $\mathbf{g} = \text{vec}(\mathbf{G}) \in \mathbb{R}^J$ ($J = \prod_{n=1}^N J_n$) and $\mathbf{a} = \text{vec}(\mathbf{A}) \in \mathbb{R}^I$ ($I = \prod_{n=1}^N I_n$) are vectors yielded by arranging the mode-1 fibers of the tensors \mathbf{G} and \mathbf{A} . The Kronecker product is denoted by symbol " \otimes ."

Moreover, given the tensor \mathbf{A} , $\|\mathbf{A}\|_0$ represents the ℓ_0 -norm which equals the number of nonzero elements of \mathbf{A} , $\|\mathbf{A}\|_1 = \sum_{i_1, i_2, \dots, i_N} |a_{i_1 i_2, \dots, i_N}|$ denotes the ℓ_1 -norm, and $\|\mathbf{A}\|_F = \sqrt{\sum_{i_1, i_2, \dots, i_N} |a_{i_1 i_2, \dots, i_N}|^2}$ denotes the Frobenius norm.

The definition of rank-one tensor is introduced at last. The N th-order tensor \mathbf{A} is rank-one if it can be written as the outer product of N vectors, i.e., $\mathbf{A} = \mathbf{a}_1 \circ \mathbf{a}_2 \circ \dots \circ \mathbf{a}_N$. The symbol " \circ " denotes the vector outer product [46].

2.1.2. Tensor Factorizations. CANDECOMP/PARAFAC (CP) factorization decomposes a tensor into a sum of component rank-one tensors [47]. For example, given a third-order tensor $\mathbf{X} \in \mathbb{R}^{I \times J \times K}$, we may formulate it as

$$\mathbf{X} \approx \sum_{r=1}^R \mathbf{a}_r \circ \mathbf{b}_r \circ \mathbf{c}_r, \quad (8)$$

where R is a positive integer and $\mathbf{a}_r \in \mathbb{R}^I$, $\mathbf{b}_r \in \mathbb{R}^J$, and $\mathbf{c}_r \in \mathbb{R}^K$ ($r = 1, 2, \dots, R$). The element of tensor \mathbf{X} can be computed by

$$x_{ijk} \approx \sum_{r=1}^R a_{ir} b_{jr} c_{kr}, \quad i = 1, \dots, I, j = 1, \dots, J, k = 1, \dots, K. \quad (9)$$

CP factorization is illustrated in Figure 1.

The factorization result can be expressed by factor matrices of three dimensions. Factor matrices refer to the combination of the vectors from the rank-one components; i.e.,

$$\begin{aligned} \mathbf{A} &= [\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_R], \\ \mathbf{B} &= [\mathbf{b}_1, \mathbf{b}_2, \dots, \mathbf{b}_R], \\ \mathbf{C} &= [\mathbf{c}_1, \mathbf{c}_2, \dots, \mathbf{c}_R]. \end{aligned} \quad (10)$$

Following [48], the CP model can be concisely represented as

$$\mathbf{X} \approx \llbracket \mathbf{A}, \mathbf{B}, \mathbf{C} \rrbracket \equiv \sum_{r=1}^R \mathbf{a}_r \circ \mathbf{b}_r \circ \mathbf{c}_r. \quad (11)$$

On the basis of factor matrices, the mode- n unfolding matrices $\mathbf{X}_{(n)}$ ($n = 1, 2, 3$) of \mathbf{X} can be represented as

$$\begin{aligned} \mathbf{X}_{(1)} &\approx \mathbf{A} (\mathbf{C} \circ \mathbf{B})^T, \\ \mathbf{X}_{(2)} &\approx \mathbf{B} (\mathbf{C} \circ \mathbf{A})^T, \\ \mathbf{X}_{(3)} &\approx \mathbf{C} (\mathbf{B} \circ \mathbf{A})^T, \end{aligned} \quad (12)$$

where the symbol " \circ " denotes the Khatri-Rao product [49]. In this way, loss functions can be modeled as the approximation of the mode- n unfolding matrices; then the factor matrices of CP factorization can be obtained by solving the corresponding optimization problem.

Tucker factorization is another popular tensor decomposing approach [50]. It decomposes a tensor into a core tensor multiplied by a matrix along each mode. Thus, in the same case as above where $\mathbf{X} \in \mathbb{R}^{I \times J \times K}$, the factorization can be described as

$$\mathbf{X} \approx \mathbf{Z} \times_1 \mathbf{A} \times_2 \mathbf{B} \times_3 \mathbf{C} = \sum_{p=1}^P \sum_{q=1}^Q \sum_{r=1}^R z_{pqr} \mathbf{a}_p \circ \mathbf{b}_q \circ \mathbf{c}_r, \quad (13)$$

where $\mathbf{A} \in \mathbb{R}^{I \times P}$, $\mathbf{B} \in \mathbb{R}^{J \times Q}$, and $\mathbf{C} \in \mathbb{R}^{K \times R}$ are factor matrices which can be regarded as the principal components in each mode. Therefore, Tucker factorization is a form of higher-

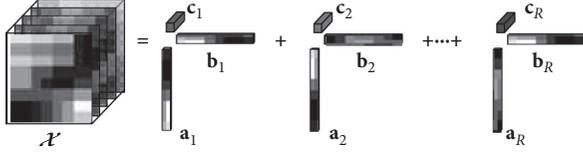


FIGURE 1: CP factorization of a third-order tensor.

order principal component analysis (PCA). Tensor $\mathbf{Z} \in \mathbb{R}^{P \times Q \times R}$ is the core tensor and its elements stand for the correlation level between the different components. Similar to (11), the Tucker model can be concisely represented by $\mathbf{X} \approx \llbracket \mathbf{Z}; \mathbf{A}, \mathbf{B}, \mathbf{C} \rrbracket$. Elementwise equation (13) can be represented as

$$x_{ijk} \approx \sum_{p=1}^P \sum_{q=1}^Q \sum_{r=1}^R z_{pqr} a_{ip} b_{jq} c_{kr}, \quad i = 1, \dots, I, j = 1, \dots, J, k = 1, \dots, K. \quad (14)$$

The Tucker factorization is illustrated in Figure 2.

2.2. The Proposed CTCF-Based Detection Method. In this subsection, the optimization problem of updating factor matrix is presented, followed with the proposed cumulative tensor CP factorization (CTCF) of third-order tensors. It is then extended to N th-order tensors. The CTCF-based detection method is described in the end of this subsection with its flowchart shown in Figure 3.

2.2.1. CP Tensor Approximation by Factor Matrices. Similar to equation (12), the mode- n unfolding matrix of $\mathbf{X} \in \mathbb{R}^{I_1 \times I_2 \times \dots \times I_N}$ can be approximated by factor matrices; i.e.,

$$\begin{aligned} \mathbf{X}_{(n)} &\approx \mathbf{A}^{(n)} \left(\mathbf{A}^{(N)} \odot \dots \odot \mathbf{A}^{(n+1)} \odot \mathbf{A}^{(n-1)} \odot \dots \odot \mathbf{A}^{(1)} \right)^T \\ &= \mathbf{A}^{(n)} \left(\odot_{i \neq n}^N \mathbf{A}^{(i)} \right)^T, \end{aligned} \quad (15)$$

where the factor matrices $\mathbf{A}^{(1)}, \dots, \mathbf{A}^{(N)}$ are obtained by CP factorization. The corresponding loss function is

$$\mathcal{L} = \frac{1}{2} \left\| \mathbf{X}_{(n)} - \mathbf{A}^{(n)} \left(\odot_{i \neq n}^N \mathbf{A}^{(i)} \right)^T \right\|^2. \quad (16)$$

The Alternating Least Squares (ALS) algorithm is often applied to obtain factor matrices by solving the following optimization problem:

$$\mathbf{A}^{(n)} = \arg \min_{\mathbf{A}^{(n)}} \frac{1}{2} \left\| \mathbf{X}_{(n)} - \mathbf{A}^{(n)} \left(\odot_{i \neq n}^N \mathbf{A}^{(i)} \right)^T \right\|^2. \quad (17)$$

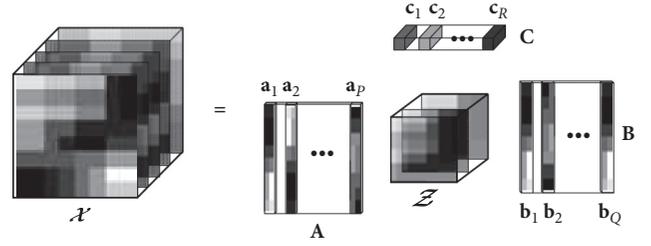


FIGURE 2: Tucker factorization of a third-order tensor.

When the tensor updates, the new tensor can be computed by the updated factor matrices which are given by equation (17).

2.2.2. CTCF of Third-Order Tensor. Generally, an image is a second-order tensor; then sequential images form a third-order tensor, i.e., a video, adding a temporal dimension on two spatial dimensions. When a new video frame presents and is added to the time dimension of the original tensor, it is defined as a three-dimensional (3D) cumulative tensor. With the number of new frames increasing, the 3D cumulative tensor updates frame by frame.

In conventional CP tensor approximation, whenever a new frame of image is added in the time dimension, ALS algorithm needs to be reused to approximate the new cumulative tensor, which is a time consuming process. In addition, the temporal correlation between neighboring frames is not exploited in the decomposition of the cumulative tensor. This paper proposes CTCF to update the CP factorization of original cumulative tensor, obtain the updated factor matrices, and approximate the new frame.

Given an original 3D cumulative tensor $\mathbf{X}_{\text{ori}} \in \mathbb{R}^{I \times J \times T_{\text{ori}}}$, the result of CP factorization is denoted by $\mathbf{X}_{\text{ori}} \approx \llbracket \mathbf{A}_{\text{ori}}, \mathbf{B}_{\text{ori}}, \mathbf{C}_{\text{ori}} \rrbracket$. When a new tensor $\mathbf{X}_{\text{new}} \in \mathbb{R}^{I \times J \times T_{\text{new}}}$ is added in the time dimension, the updated cumulative tensor is $\mathbf{X} \in \mathbb{R}^{I \times J \times (T_{\text{ori}} + T_{\text{new}})}$, of which the CP factorization appears as $\mathbf{X} \approx \llbracket \mathbf{A}, \mathbf{B}, \mathbf{C} \rrbracket$. We focus on obtaining \mathbf{A} , \mathbf{B} , and \mathbf{C} by updating \mathbf{A}_{ori} , \mathbf{B}_{ori} , and \mathbf{C}_{ori} .

The updating process is operated in an alternating way. Firstly, temporal dimensional factor matrix \mathbf{C} is computed while factor matrices \mathbf{A} and \mathbf{B} are fixed; i.e.,

$$\begin{aligned} \mathbf{C} &= \arg \min_{\mathbf{C}} \frac{1}{2} \left\| \mathbf{X}_{(3)} - \mathbf{C} (\mathbf{B} \odot \mathbf{A})^T \right\|^2 = \arg \min_{\mathbf{C}} \frac{1}{2} \left\| \begin{bmatrix} \mathbf{X}_{\text{ori}(3)} \\ \mathbf{X}_{\text{new}(3)} \end{bmatrix} - \begin{bmatrix} \mathbf{C}^{(1)} \\ \mathbf{C}^{(2)} \end{bmatrix} (\mathbf{B} \odot \mathbf{A})^T \right\|^2 \\ &= \arg \min_{\mathbf{C}} \frac{1}{2} \left\| \begin{bmatrix} \mathbf{X}_{\text{ori}(3)} - \mathbf{C}^{(1)} (\mathbf{B} \odot \mathbf{A})^T \\ \mathbf{X}_{\text{new}(3)} - \mathbf{C}^{(2)} (\mathbf{B} \odot \mathbf{A})^T \end{bmatrix} \right\|^2, \end{aligned} \quad (18)$$

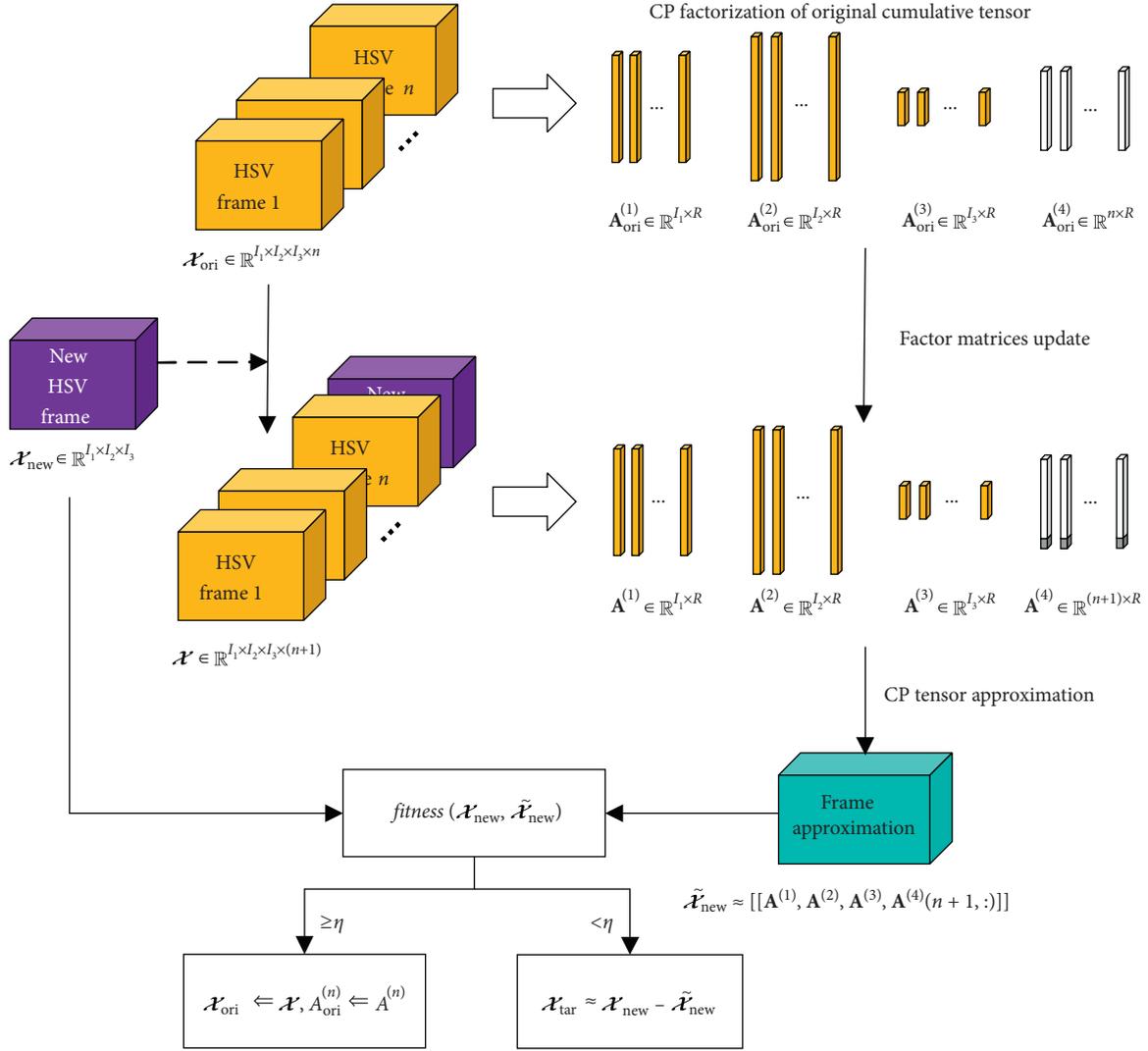


FIGURE 3: Flowchart of CTCF-based detection method.

where \mathbf{C} is divided into two terms. For \mathbf{A} and \mathbf{B} are fixed as \mathbf{A}_{ori} and \mathbf{B}_{ori} , the first row of (18) will be minimized if $\mathbf{C}^{(1)} = \mathbf{C}_{\text{ori}}$. To minimize the second row, according to (12), the optimal solution of $\mathbf{C}^{(2)}$ is $\mathbf{X}_{\text{new}(3)} ((\mathbf{B} \odot \mathbf{A})^T)^\dagger$, where the symbol “ \dagger ” denotes Moore–Penrose pseudoinverse of the matrix [51]. So, \mathbf{C} can be updated by adding \mathbf{C}_{new} which is represented by

$$\mathbf{C} = \begin{bmatrix} \mathbf{C}_{\text{ori}} \\ \mathbf{C}_{\text{new}} \end{bmatrix} = \begin{bmatrix} \mathbf{C}_{\text{ori}} \\ \mathbf{X}_{\text{new}(3)} ((\mathbf{B} \odot \mathbf{A})^T)^\dagger \end{bmatrix}. \quad (19)$$

Secondly, factor matrix \mathbf{A} is computed while factor matrices \mathbf{B} and \mathbf{C} are fixed. Similar to 16, the loss function of estimating \mathbf{A} is written as

$$\mathcal{L} = \frac{1}{2} \|\mathbf{X}_{(1)} - \mathbf{A}(\mathbf{C} \odot \mathbf{B})^T\|^2. \quad (20)$$

Derive \mathcal{L} with respect to \mathbf{A} ; then, we have

$$\frac{\partial \mathcal{L}}{\partial \mathbf{A}} = \mathbf{X}_{(1)} (\mathbf{C} \odot \mathbf{B}) - \mathbf{A} (\mathbf{C} \odot \mathbf{B})^T (\mathbf{C} \odot \mathbf{B}). \quad (21)$$

To simplify equation (21), denote $\mathbf{P} = \mathbf{X}_{(1)} (\mathbf{C} \odot \mathbf{B})$ and $\mathbf{Q} = (\mathbf{C} \odot \mathbf{B})^T (\mathbf{C} \odot \mathbf{B})$; thus, when $\partial \mathcal{L} / \partial \mathbf{A} = 0$, we have $\mathbf{A} = \mathbf{P} \mathbf{Q}^{-1}$. According to [47], \mathbf{Q} can be rewritten as

$$\mathbf{Q} = (\mathbf{C} \odot \mathbf{B})^T (\mathbf{C} \odot \mathbf{B}) = (\mathbf{C}^T \mathbf{C}) \otimes (\mathbf{B}^T \mathbf{B}). \quad (22)$$

For computing \mathbf{P} , we also divide $\mathbf{X}_{(1)}$ and \mathbf{C} into two terms; i.e.,

$$\begin{aligned} \mathbf{P} &= \mathbf{X}_{(1)} (\mathbf{C} \odot \mathbf{B}) = [\mathbf{X}_{\text{ori}(1)}, \mathbf{X}_{\text{new}(1)}] \left(\begin{bmatrix} \mathbf{C}_{\text{ori}} \\ \mathbf{C}_{\text{new}} \end{bmatrix} \odot \mathbf{B} \right) \\ &= [\mathbf{X}_{\text{ori}(1)}, \mathbf{X}_{\text{new}(1)}] \left(\begin{bmatrix} \mathbf{C}_{\text{ori}} \odot \mathbf{B} \\ \mathbf{C}_{\text{new}} \odot \mathbf{B} \end{bmatrix} \right) = \mathbf{X}_{\text{ori}(1)} (\mathbf{C}_{\text{ori}} \odot \mathbf{B}) + \mathbf{X}_{\text{new}(1)} (\mathbf{C}_{\text{new}} \odot \mathbf{B}). \end{aligned} \quad (23)$$

Since \mathbf{B} are fixed as \mathbf{B}_{ori} , the first term of equation (23) contains only the information of original tensor, which can be expressed by

$$\mathbf{P}_{\text{ori}} = \mathbf{X}_{\text{ori}(1)} (\mathbf{C}_{\text{ori}} \odot \mathbf{B}), \quad (24)$$

so, equation (23) is rewritten as

$$\mathbf{P} = \mathbf{P}_{\text{ori}} + \mathbf{X}_{\text{new}(1)} (\mathbf{C}_{\text{new}} \odot \mathbf{B}), \quad (25)$$

Hence, \mathbf{P} can be updated from \mathbf{P}_{ori} using mode-1 unfolding matrix of \mathbf{X}_{new} and factor matrix mentioned above \mathbf{C}_{new} . Generally, \mathbf{P} is initialized by $\mathbf{X}(\tau) \in \mathbb{R}^{I \times J \times \tau}$, which is a small front part of \mathbf{X}_{ori} , and updated iteratively by (25). Analogously, the update process of \mathbf{Q} can be represented by

$$\mathbf{Q} = \mathbf{Q}_{\text{ori}} + (\mathbf{C}_{\text{new}} \odot \mathbf{B})^T (\mathbf{C}_{\text{new}} \odot \mathbf{B}) = \mathbf{Q}_{\text{ori}} + (\mathbf{C}_{\text{new}}^T \mathbf{C}_{\text{new}}) \otimes (\mathbf{B}^T \mathbf{B}). \quad (26)$$

The update of \mathbf{A} may be summarized as

$$\begin{aligned} \mathbf{P} &\leftarrow \mathbf{P} + \mathbf{X}_{\text{new}(1)} (\mathbf{C}_{\text{new}} \odot \mathbf{B}), \\ \mathbf{Q} &\leftarrow \mathbf{Q} + (\mathbf{C}_{\text{new}}^T \mathbf{C}_{\text{new}}) \otimes (\mathbf{B}^T \mathbf{B}), \\ \mathbf{A} &\leftarrow \mathbf{P} \mathbf{Q}^{-1}. \end{aligned} \quad (27)$$

Finally, the update of factor matrix \mathbf{B} may likewise be expressed by

$$\begin{aligned} \mathbf{U} &\leftarrow \mathbf{U} + \mathbf{X}_{\text{new}(2)} (\mathbf{C}_{\text{new}} \odot \mathbf{A}), \\ \mathbf{V} &\leftarrow \mathbf{V} + (\mathbf{C}_{\text{new}}^T \mathbf{C}_{\text{new}}) \otimes (\mathbf{A}^T \mathbf{A}), \\ \mathbf{B} &\leftarrow \mathbf{U} \mathbf{V}^{-1}, \end{aligned} \quad (28)$$

where $\mathbf{U} = \mathbf{X}_{(2)} (\mathbf{C} \odot \mathbf{A})$ and $\mathbf{V} = (\mathbf{C}^T \mathbf{C}) \otimes (\mathbf{A}^T \mathbf{A})$.

To make the process clearer, the proposed CTCF of third-order tensor is summarized by Algorithm 1.

2.2.3. CTCF of Nth-Order Tensor. On the basis of Section 2.2.2, we try to extend CTCF to higher-order tensors. Suppose an N -dimensional cumulative tensor $\mathbf{X}_{\text{ori}} \in \mathbb{R}^{I_1 \times I_2 \times \dots \times I_{N-1} \times T_{\text{ori}}}$ where the last dimension is temporal dimension. The CP factorization of \mathbf{X}_{ori} is represented as $\mathbf{X}_{\text{ori}} \approx [\mathbf{A}_{\text{ori}}^{(1)}, \dots, \mathbf{A}_{\text{ori}}^{(N-1)}, \mathbf{A}_{\text{ori}}^{(N)}]$. When a new tensor $\mathbf{X}_{\text{ori}} \in \mathbb{R}^{I_1 \times I_2 \times \dots \times I_{N-1} \times T_{\text{new}}}$ is added in the time dimension, the updated cumulative tensor is $\mathbf{X} \in \mathbb{R}^{I_1 \times I_2 \times \dots \times I_{N-1} \times (T_{\text{ori}} + T_{\text{new}})}$, of which the CP factorization is denoted by $\mathbf{X} \approx [\mathbf{A}^{(1)}, \dots, \mathbf{A}^{(N-1)}, \mathbf{A}^{(N)}]$.

Similar to Section 2.2.2, temporal dimensional factor matrix $\mathbf{A}^{(N)}$ is firstly updated with other $N-1$ matrices fixed. Like 17, the optimization problem of estimating $\mathbf{A}^{(N)}$ is formulated by

$$\mathbf{A}^{(N)} = \arg \min_{\mathbf{A}^{(N)}} \frac{1}{2} \left\| \mathbf{X}_{(N)} - \mathbf{A}^{(N)} (\odot^{N-1} \mathbf{A}^{(i)})^T \right\|^2. \quad (29)$$

We also separate original part from new added part; i.e.,

$$\mathbf{A}^{(N)} = \begin{bmatrix} \mathbf{A}_{\text{ori}}^{(N)} \\ \mathbf{A}_{\text{new}}^{(N)} \end{bmatrix} = \begin{bmatrix} \mathbf{A}_{\text{ori}}^{(N)} \\ \mathbf{X}_{\text{new}} \left((\odot^{N-1} \mathbf{A}^{(i)})^T \right)^\dagger \end{bmatrix}, \quad i = 1, 2, \dots, N-1. \quad (30)$$

The original part is minimized by fixing the first $N-1$ factor matrix and the new part is updated by \mathbf{X}_{new} .

The updates of nontemporal dimensional factor matrices $\mathbf{A}^{(n)}$ ($n \in [1, N-1]$) may refer to the ones of factor matrices \mathbf{A} and \mathbf{B} in Section 2.2.2. The loss function $\mathcal{L}^{(n)}$ of estimating $\mathbf{A}^{(n)}$ is the same as 16. Let $\partial \mathcal{L}^{(n)} / \partial \mathbf{A}^{(n)} = 0$ and introduce matrices $\mathbf{P}^{(n)}$ and $\mathbf{Q}^{(n)}$; the update of $\mathbf{A}^{(n)}$ may be summarized as

$$\begin{aligned} \mathbf{P}^{(n)} &\leftarrow \mathbf{P}^{(n)} + \mathbf{X}_{\text{new}(n)} (\mathbf{A}_{\text{new}}^{(N)} \odot \mathbf{K}^{(n)}), \\ \mathbf{Q}^{(n)} &\leftarrow \mathbf{Q}^{(n)} + (\mathbf{A}_{\text{new}}^{(N)T} \mathbf{A}_{\text{new}}^{(N)}) \otimes \mathbf{R}^{(n)}, \\ \mathbf{A}^{(n)} &\leftarrow \mathbf{P}^{(n)} (\mathbf{Q}^{(n)})^{-1}, \end{aligned} \quad (31)$$

where $\mathbf{K}^{(n)} = \odot_{i \neq n}^{N-1} \mathbf{A}^{(i)}$ and $\mathbf{R}^{(n)} = \otimes_{i \neq n}^{N-1} \mathbf{A}^{(i)T} \mathbf{A}^{(i)}$.

2.2.4. CTCF-Based Detection Method. In HSV, the sequential data is expressed as a 4D cumulative tensor; the temporal dimension increases with new frames are added in. Whenever a new frame presents, the results of original cumulative tensor CP factorization are updated to obtain the factor matrices of the new cumulative tensor, and the CP tensor approximation of the newly added frame is obtained at the same time. If the target is absent, the CP tensor approximation will lead to a small error, since the background information is similar between adjacent frames. On the contrary, if the error is large, the target is likely to present. We define the fitness between the new frame and its approximation in 34. If the fitness is smaller than the threshold, the target is supposed to appear in the new frame. Otherwise, the new frame is added in the temporal dimension and used to update original cumulative tensor.

The original 4D cumulative tensor is denoted by $\mathbf{X}_{\text{ori}} \in \mathbb{R}^{I_1 \times I_2 \times I_3 \times n}$; n denotes the frame number of initial video. The factor matrices of four dimensions are represented as

$$\mathbf{X}_{\text{ori}} \approx [\mathbf{A}_{\text{ori}}^{(1)}, \mathbf{A}_{\text{ori}}^{(2)}, \mathbf{A}_{\text{ori}}^{(3)}, \mathbf{A}_{\text{ori}}^{(4)}], \quad (32)$$

where $\mathbf{A}_{\text{ori}}^{(1)} \in \mathbb{R}^{I_1 \times R}$, $\mathbf{A}_{\text{ori}}^{(2)} \in \mathbb{R}^{I_2 \times R}$, $\mathbf{A}_{\text{ori}}^{(3)} \in \mathbb{R}^{I_3 \times R}$, and $\mathbf{A}_{\text{ori}}^{(4)} \in \mathbb{R}^{n \times R}$ and R denotes the number of component rank-one tensors in CP factorization. When a new frame $\mathbf{X}_{\text{new}} \in \mathbb{R}^{I_1 \times I_2 \times I_3}$ is added in the temporal dimension of

Input: original 3D cumulative tensor $\mathbf{X}_{\text{ori}} \in \mathbb{R}^{I \times J \times T_{\text{ori}}} \approx [\mathbf{A}_{\text{ori}}, \mathbf{B}_{\text{ori}}, \mathbf{C}_{\text{ori}}]$ new tensor $\mathbf{X}_{\text{new}} \in \mathbb{R}^{I \times J \times T_{\text{new}}}$
 Step 1: new tensor is added in the time dimension and $\mathbf{X} \in \mathbb{R}^{I \times J \times (T_{\text{ori}} + T_{\text{new}})}$ is obtained
 Step 2: decompose \mathbf{X} by CP factorization $\mathbf{X} \approx [\mathbf{A}, \mathbf{B}, \mathbf{C}]$
 Step 3: update \mathbf{C} by (19), with \mathbf{A} and \mathbf{B} are fixed
 Step 4: update \mathbf{A} by (27), with \mathbf{B} and \mathbf{C} are fixed
 Step 5: update \mathbf{B} by (28), with \mathbf{A} and \mathbf{C} are fixed
 Step 6: estimate \mathbf{X} by updated \mathbf{A} , \mathbf{B} and \mathbf{C}
Output: approximation of updated cumulative tensor \mathbf{X}

ALGORITHM 1: CTCF of third-order tensor.

original 4D cumulative tensor, the 4D cumulative tensor is updated and denoted by $\mathbf{X} \in \mathbb{R}^{I_1 \times I_2 \times I_3 \times (n+1)}$. The factor matrices of \mathbf{X} are expressed by

$$\mathbf{X} \approx [\mathbf{A}^{(1)}, \mathbf{A}^{(2)}, \mathbf{A}^{(3)}, \mathbf{A}^{(4)}], \quad (33)$$

where $\mathbf{A}^{(1)} \in \mathbb{R}^{I_1 \times R}$, $\mathbf{A}^{(2)} \in \mathbb{R}^{I_2 \times R}$, $\mathbf{A}^{(3)} \in \mathbb{R}^{I_3 \times R}$, and $\mathbf{A}^{(4)} \in \mathbb{R}^{(n+1) \times R}$. Based on Section 2.2.3, we estimate $\mathbf{A}^{(1)} \sim \mathbf{A}^{(4)}$ and obtain the approximation of \mathbf{X} and $\tilde{\mathbf{X}}_{\text{new}}$, where $\tilde{\mathbf{X}}_{\text{new}} \approx [\mathbf{A}^{(1)}, \mathbf{A}^{(2)}, \mathbf{A}^{(3)}, \mathbf{A}^{(4)}(n+1, :)]$. Actually, it is the specific case when $N = 4$.

We define the *fitness* ($\mathbf{X}_{\text{new}}, \tilde{\mathbf{X}}_{\text{new}}$) as

$$\text{fitness}(\mathbf{X}_{\text{new}}, \tilde{\mathbf{X}}_{\text{new}}) = 1 - \frac{\|\mathbf{X}_{\text{new}} - \tilde{\mathbf{X}}_{\text{new}}\|_F}{\|\mathbf{X}_{\text{new}}\|_F}. \quad (34)$$

If the target does not appear, the approximation error is small and the result of fitness is large. Given a preset threshold η , when $\text{fitness}(\mathbf{X}_{\text{new}}, \tilde{\mathbf{X}}_{\text{new}}) > \eta$, i.e., the fitness is larger than η , we decide that the target is absent. Then, the nontarget frame is added in temporal dimension and the updated 4D cumulative tensor becomes the new original 4D cumulative tensor, which can be expressed as

$$\mathbf{X}_{\text{ori}} \leftarrow \mathbf{X}, \mathbf{A}_{\text{ori}}^{(n)} \leftarrow \mathbf{A}^{(n)}, \quad n \in [1, 4]. \quad (35)$$

If the target appears, the approximation error is large and the fitness is smaller than η . The residual of \mathbf{X}_{new} and $\tilde{\mathbf{X}}_{\text{new}}$ is the approximation of the target tensor; i.e.,

$$\mathbf{X}_{\text{tar}} \approx \mathbf{X}_{\text{new}} - \tilde{\mathbf{X}}_{\text{new}}. \quad (36)$$

The target of each frame will be shown in 2D form by taking the maximum value of every spectrum. In this way, the proposed CTCF-based detection method can extract not only the key-frames where the target presents, but also the approximate region of target in every key-frame. The flowchart of the proposed method is shown in Figure 3. In Section 3, experiments on real HSV data are conducted and the proposed method is compared with some representative techniques.

2.3. The Proposed STTF-Based Super-Resolution Method. In Section 2.2, we present an approach to detect the frames where the target appears in HSV and the approximate region of the target. However, as discussed in Section 1, there has to be a tradeoff between spectral resolution and the spatial resolution in HSI imaging systems [52]. The spatial resolution is always low since high spectral resolution is required

in HSIs and HSV. So, we are interested in improving the spatial resolution of targets after the detecting process. Instead of fusing HR-MSI and LR-HSI, we try to handle the target SR problem by what we have got, which is more practical in real cases.

2.3.1. Problem Formulation. In this subsection, HSIs are represented as 3D tensors with three indexes (H, W, S), which stand for the height, width, and spectral modes. $\mathbf{X} \in \mathbb{R}^{H \times W \times S}$ denotes the HR-HSI and the LR-HSI is denoted by $\mathbf{Y} \in \mathbb{R}^{h \times w \times S}$, where $W > w$ and $H > h$. The goal is to estimate \mathbf{X} from \mathbf{Y} .

There are two significant characteristics of HR-HSIs [53]: the first one is that spectral vectors can be well approximated in low dimensional subspaces, and the second one is that HSIs are spatially self-similar. This means that sparsity exists in both spectral and spatial dimensions. Inspired by sparse representation [54], the low dimensionality in spectral domain gives the possibility to form a spectral mode dictionary \mathbf{S} with few nonzero atoms; the self-similarities in spatial domain guarantee the sparse representations of the height and width modes with spatial dictionaries \mathbf{H} and \mathbf{W} . In this way, the conventional Tucker factorization is transformed into the multiplication of the core tensor and three modes dictionaries. The factorization is illustrated in Figure 4. The HR-HSI is represented as

$$\mathbf{X} = \mathbf{Z} \times_1 \mathbf{H} \times_2 \mathbf{W} \times_3 \mathbf{S}, \quad (37)$$

where $\mathbf{H} \in \mathbb{R}^{H \times z_h}$, $\mathbf{W} \in \mathbb{R}^{W \times z_w}$, and $\mathbf{S} \in \mathbb{R}^{S \times z_s}$. The variables z_h , z_w , and z_s denote the atoms (i.e., the number of columns) of \mathbf{H} , \mathbf{W} , and \mathbf{S} , respectively. The core tensor \mathbf{Z} contains the coefficients of \mathbf{X} over three modes dictionaries. We can see that 37 incorporates the information of separated modes into a unified framework.

The LR key-frame of HSV \mathbf{Y} can be seen as the spatially downsampled version of HR-HSI \mathbf{X} , which is written as

$$\mathbf{Y} = \mathbf{X} \times_1 \mathbf{D}_1 \times_2 \mathbf{D}_2, \quad (38)$$

where $\mathbf{D}_1 \in \mathbb{R}^{h \times H}$ and $\mathbf{D}_2 \in \mathbb{R}^{w \times W}$ are downsampling matrices of the height and width modes. Substituting 37 into (38), \mathbf{Y} is represented by

$$\mathbf{Y} = \mathbf{Z} \times_1 (\mathbf{D}_1 \mathbf{H}) \times_2 (\mathbf{D}_2 \mathbf{W}) \times_3 \mathbf{S} = \mathbf{Z} \times_1 \mathbf{H}^* \times_2 \mathbf{W}^* \times_3 \mathbf{S}, \quad (39)$$

where $\mathbf{H}^* = \mathbf{D}_1 \mathbf{H} \in \mathbb{R}^{h \times z_h}$ and $\mathbf{W}^* = \mathbf{D}_2 \mathbf{W} \in \mathbb{R}^{w \times z_w}$ denotes the downsampled dictionary of height and width modes. To

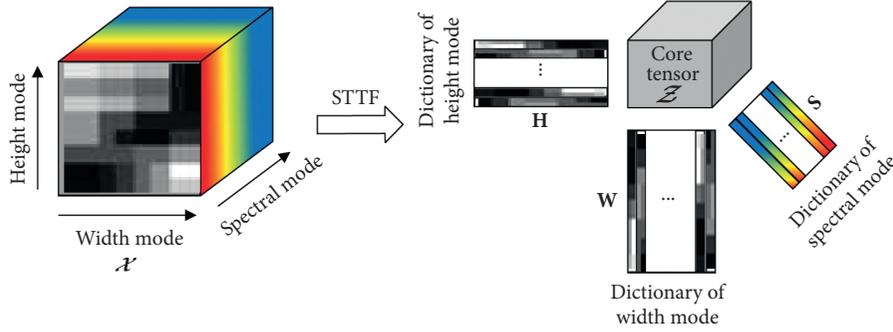


FIGURE 4: Sparse-based tensor Tucker factorization of an HSI.

recover \mathbf{X} , we focus on estimating the dictionaries \mathbf{H} , \mathbf{W} , and \mathbf{S} and the core tensor \mathbf{Z} .

2.3.2. The Proposed STTF-Based SR Algorithm. Since \mathbf{Y} is a downsampled version, recovering \mathbf{X} from \mathbf{Y} is a typical inverse problem, which is badly ill-posed. So, some prior knowledge of \mathbf{X} is needed to regularize the super-resolution problem. In HSI processing, the spectral sparsity is a widespread regularizer applied to solve varieties of ill-posed problems [55–58]. In such regularization, spectral vectors are linearly combined by a small quantity of different spectral signatures. However, these schemes only take advantage of the sparsity of the spectral domain. In the proposed algorithm, taking into account the HSI self-similarity, sparsity regularization is extended to the spatial domain by exploiting the sparse-based tensor Tucker factorization (STTF). In STTF, the HR-HSI performs a united sparse representation of the core tensor and three modes dictionaries.

On the basis of equation (39), the HSV frame super-resolution is formulated as a constrained least-squares optimization problem:

$$\begin{aligned} \min_{\mathbf{H}, \mathbf{W}, \mathbf{S}, \mathbf{Z}} \|\mathbf{Y} - \mathbf{Z} \times_1 \mathbf{H}^* \times_2 \mathbf{W}^* \times_3 \mathbf{S}\|_F^2, \\ \text{s.t. } \|\mathbf{Z}\|_0 \leq \theta, \end{aligned} \quad (40)$$

where $\|\cdot\|_F$ represents the Frobenius norm and θ denotes the number of maximum nonzero elements of \mathbf{Z} . Because of the ℓ_0 -norm constraint, equation (40) is nonconvex. To make the optimization processable, the ℓ_0 -norm is replaced by the ℓ_1 -norm and 40 is transformed into an unconstrained version:

$$\min_{\mathbf{H}, \mathbf{W}, \mathbf{S}, \mathbf{Z}} \|\mathbf{Y} - \mathbf{Z} \times_1 \mathbf{H}^* \times_2 \mathbf{W}^* \times_3 \mathbf{S}\|_F^2 + \lambda \|\mathbf{Z}\|_1, \quad (41)$$

where λ is the parameter of sparse regularizer. Equation (41) is also nonconvex, and the solutions of \mathbf{H} , \mathbf{W} , and \mathbf{S} and \mathbf{Z} are not unique. Nonetheless, if we focus on only one variable with other variables fixed, the objective function in equation (41) is convex. Inspired by [59, 60], equation (41) can be solved by proximal alternating optimization scheme, which is guaranteed to reach convergence in a particular situation. Concretely, \mathbf{H} , \mathbf{W} , \mathbf{S} , and \mathbf{Z} are updated iteratively by

$$\begin{aligned} \mathbf{H} &= \arg \min_{\mathbf{H}} f(\mathbf{H}, \mathbf{W}, \mathbf{S}, \mathbf{Z}) + \alpha \|\mathbf{H} - \mathbf{H}_{\text{pre}}\|_F^2, \\ \mathbf{W} &= \arg \min_{\mathbf{W}} f(\mathbf{H}, \mathbf{W}, \mathbf{S}, \mathbf{Z}) + \alpha \|\mathbf{W} - \mathbf{W}_{\text{pre}}\|_F^2, \\ \mathbf{S} &= \arg \min_{\mathbf{S}} f(\mathbf{H}, \mathbf{W}, \mathbf{S}, \mathbf{Z}) + \alpha \|\mathbf{S} - \mathbf{S}_{\text{pre}}\|_F^2, \\ \mathbf{Z} &= \arg \min_{\mathbf{Z}} f(\mathbf{H}, \mathbf{W}, \mathbf{S}, \mathbf{Z}) + \alpha \|\mathbf{Z} - \mathbf{Z}_{\text{pre}}\|_F^2, \end{aligned} \quad (42)$$

where $(\cdot)_{\text{pre}}$ denotes the previous estimation in the last iteration and α denotes a positive number. Equation (41) defines the object function $f(\mathbf{H}, \mathbf{W}, \mathbf{S}, \mathbf{Z})$. The optimizations of \mathbf{H} , \mathbf{W} , \mathbf{S} , and \mathbf{Z} will be presented detailedly in the appendix. The conjugate gradient (CG) method [61] and the alternating direction method of multipliers (ADMM) [62] will be used in the optimizations.

2.3.3. Initialization of the Proposed Method. Since the optimization problem in (41) is nonconvex, the solution would result in poor local minima if we set the initialization carelessly. In this paper, we initialize the spatial dictionaries \mathbf{H}^* and \mathbf{W}^* from $\mathbf{Y}_{(1)}$ and $\mathbf{Y}_{(2)}$ dictionary-updates-cycles KSVd (DUC-KSVd) [63]; this method can promote sparse representations. Then, initialization of spectral dictionary \mathbf{S} is accomplished by simplex identification split augmented Lagrangian (SISAL) algorithm [64]; this approach can efficiently identify a minimum unit that contains the spectral vectors.

The proposed STTF-based SR algorithm is summarized in Algorithm 2.

3. Results and Discussion

3.1. Experimental Data Set. To highlight the advantages of HSIs, we choose invisible gas plume to be the target. The proposed algorithms can be extended to other types of data reasonably. In this section, the HSV data set is acquired by the infrared imaging spectrometer “HyperCam-LW.” Sulfur hexafluoride (SF_6) is chosen to be the target, since it is a kind of odorless and colorless gas plume with a distinct absorption peak in LWIR range. The HSV data set consists of 60 infrared hyperspectral frames with the size of

```

Input: LR-HSI  $\mathbf{Y}$ 
Initialize  $\mathbf{S}$  with SISAL
Initialize  $\mathbf{H}^*$  and  $\mathbf{W}^*$  with DUC-KSVD
Initialize  $\mathbf{Z}$  with (39)
while no convergence do
  Step 1: update  $\mathbf{H}$  by solving (A.3) with CG
   $\mathbf{H}^* = \mathbf{D}_1 \mathbf{H}, \mathbf{H}_{\text{pre}} = \mathbf{H}$ 
  Step 2: update  $\mathbf{W}$  by solving (A.6) with CG
   $\mathbf{W}^* = \mathbf{D}_2 \mathbf{W}, \mathbf{W}_{\text{pre}} = \mathbf{W}$ 
  Step 3: update  $\mathbf{S}$  by solving (A.9) with CG
   $\mathbf{S}_{\text{pre}} = \mathbf{S}$ 
  Step 4: update  $\mathbf{Z}$  by solving (A.15) with CG
   $\mathbf{Z}_{\text{pre}} = \mathbf{Z}$ 
end while
Estimate  $\mathbf{X}$  by (37)
Output: HR-HSI  $\mathbf{X}$ 

```

ALGORITHM 2: STTF-based SR method.

$128 \times 320 \times 127$. The imaging interval is 4.8 s, and the wavelength of the data ranges from $7.8 \mu\text{m}$ to $11.8 \mu\text{m}$.

In SR method, only the middle 128×128 pixels are used in the experiment (specifically, column 71 to column 198) for reasons connected with the algorithm process. And we remove the spectral band 41–127 because of water vapor absorption and extremely low SNR. At last, the size of input LR-HSI is $128 \times 128 \times 40$.

3.2. Compared Methods. For CTCF-based detection method, we compare it with two representative methods: MSD (matched subspace detector) [25] and CMF (constrained matrix factorization) [29]. For STTF-based SR method, we compare it with three state-of-the-art algorithms: bicubic interpolation, sparse representation-based SR method [54], and sequence information-based SR method [65].

3.3. Qualitative and Quantitative Metrics. For detection methods, receiver operating characteristic (ROC) curves [66] are used to evaluate the performance. Generally, a detector outperforms another one if the area under its ROC curve is larger [67]. As suggested in [68], the area under the ROC curve (AUC) is also calculated as a measure of performance of these detection methods. Usually, a better detector gets a higher AUC value.

For SR algorithms, since we directly process the LR-HSI, there is no original HR-HSI (i.e., the ground truth) for reference. Thus, some popular quantitative metrics are not available, such as RMSE (root-mean-square error) [69], PSNR (peak signal to noise ratio), and SAM (spectral angle mapper). In this section, entropy and average gradient are introduced to evaluate the performance of SR methods.

3.3.1. Entropy. Super-resolution aims to introduce more useful information into images, so we may measure the performance of SR methods by calculating the contained information in the experimental results. The entropy is indicated as

$$E = - \sum_{i=1}^n P(i) \log_2 P(i). \quad (43)$$

The probability of a pixel i in the image is denoted by $P(i)$ and n denotes the grey value range ($0 \sim 255$). The larger the entropy value of the image, the richer the information contained in the image.

3.3.2. Average Gradient. Another assessment to measure the performance of super-resolution is the change of the amount of detailed information in the image. We may evaluate the experimental results by average gradient, since it can reflect the ability of expressing the details and measuring the clarity of the image. The gradient increases if the greyscale level rate in one direction of the image varies quickly. The average gradient is formulated as

$$\bar{G} = \frac{1}{m \times n} \sum_{i=1}^m \sum_{j=1}^n \sqrt{\left((\partial f_{i,j} / \partial x_i)^2 + (\partial f_{i,j} / \partial y_i)^2 \right) / 2}, \quad (44)$$

where m and n denote the height and width of the image, respectively; $f_{i,j}$ denotes the greyscale value of pixel (i, j) in the image. The larger the average gradient value of the image is, the clearer the image will be.

Besides, the visual quality of output images is an important qualitative metric.

3.4. Parameters Setting. In MSD, we pick 463 spectrums of gas target and 846 spectrums of background from the 12th frame of HSV to build up the training set. The size of the target subspace and background space is 127×112 and 127×115 , respectively. In CMF, the number of endmembers is 3, the sparsity of factor matrices is 2, and number of iteration is 3. In the proposed CTCF-based method, the original cumulative tensor is obtained by ALS, the tensor rank is 3, the maximum iteration number is 100, and the reconstruction error is 10^{-8} ; in update stage, the threshold of fitness is 0.9. In the proposed STTF-based SR method, the

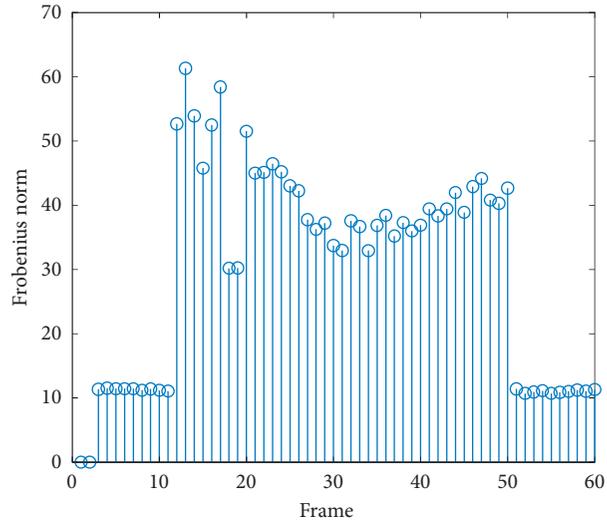


FIGURE 5: The value of Frobenius norm of each frame.

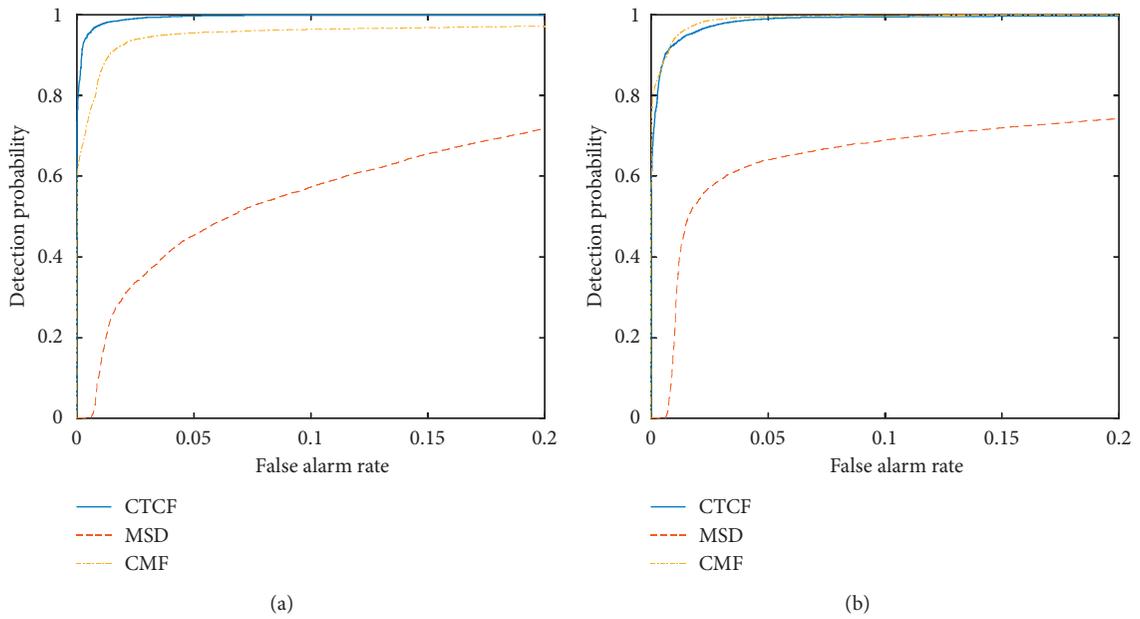


FIGURE 6: Continued.

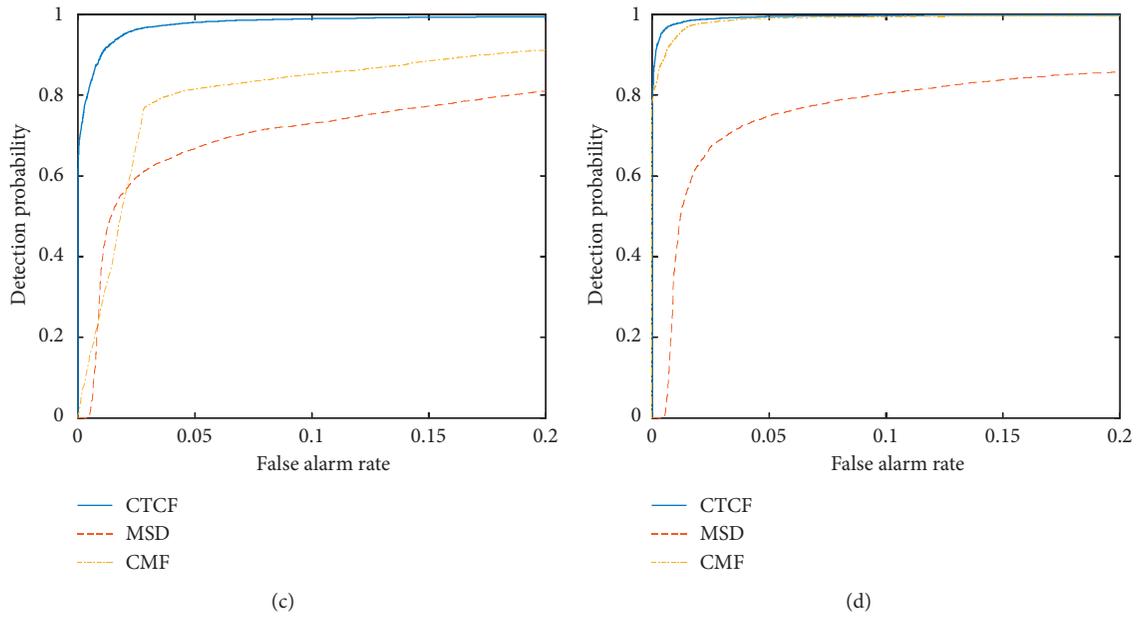


FIGURE 6: The comparison of ROC curves of three detection methods: (a) frame 13; (b) frame 23; (c) frame 34; (d) frame 44.

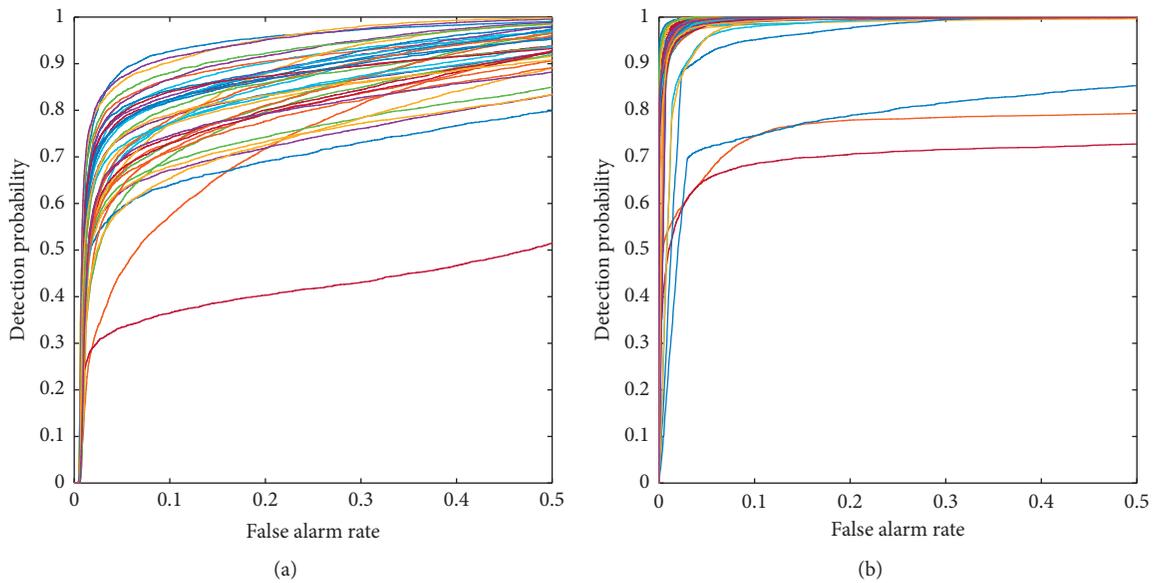


FIGURE 7: Continued.

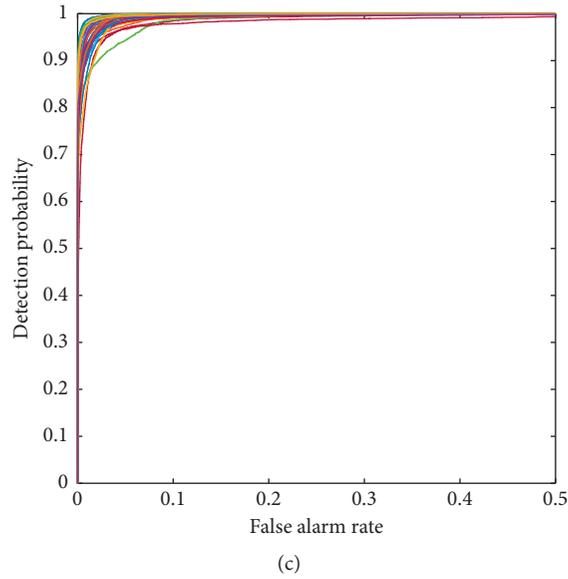


FIGURE 7: The ROC curves of 39 key-frames by three detection methods: (a) MSD; (b) CMF; (c) CTCF.

TABLE 1: Detection quantitative results (AUC value) of the test methods on key-frames.

Frame	MSD [25]	CMF [29]	CTCF
12	0.9655	0.9993	0.9980
13	0.8462	0.9995	0.9980
14	0.8878	0.9994	0.9981
15	0.8189	0.9965	0.9981
16	0.8734	0.9995	0.9977
17	0.9348	0.9987	0.9946
18	0.5792	0.7477	0.9915
19	0.7894	0.9988	0.9958
20	0.9336	0.8991	0.9934
21	0.8222	0.9980	0.9966
22	0.9001	0.9915	0.9969
23	0.8388	0.9986	0.9945
24	0.8914	0.9990	0.9983
25	0.9169	0.9989	0.9961
26	0.9254	0.9947	0.9989
27	0.8722	0.9974	0.9951
28	0.8503	0.9978	0.9930
29	0.9490	0.9892	0.9974
30	0.9011	0.9341	0.9885
31	0.9157	0.9867	0.9965
32	0.8881	0.8582	0.9811
33	0.9345	0.9771	0.9933
34	0.9007	0.9977	0.9922
35	0.9273	0.9952	0.9933
36	0.9349	0.9989	0.9950
37	0.9528	0.9986	0.9984
38	0.9299	0.9875	0.9981
39	0.8838	0.9979	0.9962
40	0.9295	0.9976	0.9966
41	0.9165	0.9939	0.9976
42	0.9665	0.9986	0.9988
43	0.9660	0.9763	0.9969
44	0.9083	0.9995	0.9979
45	0.9046	0.9977	0.9935
46	0.9156	0.9964	0.9918
47	0.9225	0.9979	0.9950

TABLE 1: Continued.

Frame	MSD [25]	CMF [29]	CTCF
48	0.8623	0.9980	0.9967
49	0.8894	0.9992	0.9980
50	0.8640	0.9969	0.9947
Average	0.8926	0.9817	0.9954
Variance	0.4407×10^{-2}	0.2290×10^{-2}	0.1134×10^{-6}

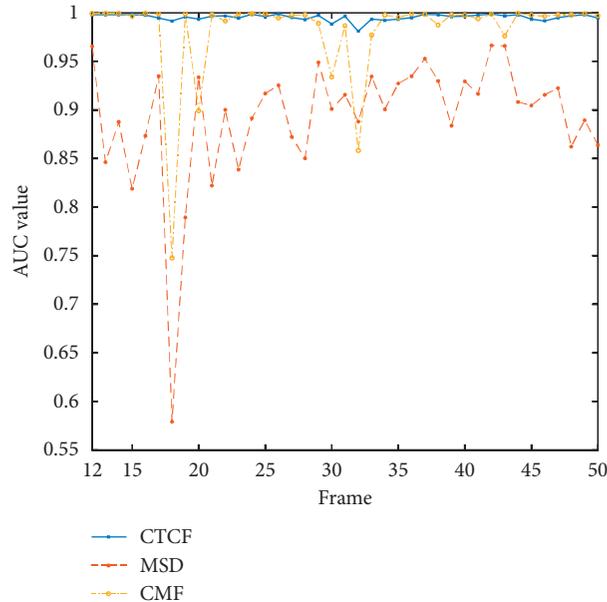


FIGURE 8: The AUC values 39 key-frames by three detection methods.

number of iterations is 5; the parameter α is the weight in (42) and we set $\alpha = 10^{-3}$; parameter λ controls the sparsity of \mathbf{Z} ; we set $\lambda = 10^{-5}$; parameter μ is set by $\mu = 10^{-2}$; the size of \mathbf{Z} is set by $z_h = 240$, $z_w = 240$, and $z_s = 12$. The parameters above are decided after sufficient number of experiments to make a balance between efficiency and stability.

3.5. Experimental Results and Discussion. In this subsection, we show the experimental results of the various methods for detection and super-resolution.

After processing the HSV by the proposed CTCF-based method, we compute the values of Frobenius norm of each frame, which are presented in Figure 5. It is obvious that target gas appears in the 12th frame and disappears in the 51st frame. Figure 6 compares the ROC curves of test methods on four frames in detail, and Figure 7 illustrates the general trends of ROC curves of MSD, CMF, and CTCF, respectively. As can be seen from Figures 6 and 7, the proposed CTCF-based detection algorithm outperforms the other two methods. The AUC values of three approaches are shown in Table 1. In each row, the bold value represents the highest AUC value. Although the AUC values of CMF in some frames are better, we can see that the AUC values of CMF in some other frames are very low (less than 0.98). On the contrast, all the results of CTCF lie in the range of 0.98 to 1. From the average value and the variance (the bold value

represents the highest value), we can conclude that the proposed method is superior and more stable. The graphical results are illustrated in Figure 8.

The target of each key-frame is shown in 2D form (grey image) by taking the maximum value of every spectrum. To save the length of the paper, we choose 8 frames to show the comparison of three detectors, which are shown in Figure 9. The first row to the eighth row present the detection result of the chosen frame, of which the frame number is 15, 18, 22, 28, 31, 39, 48, and 50. The higher the greyscale of the pixel in the image is, the closer it is to the target. It is apparent that our method extracts more accurate targets.

Table 2 shows the entropy and average gradient of the key-frames by four SR algorithms. Since sequence-based method needs 5 LR frames to form 1 HR frame, the compared frame number is changed from range 12~50 to range 14~48. In each row, the bold values represent the highest entropy value and the highest average gradient value. From Table 2, we can conclude that firstly, although interpolation can add more information in the frame, the details of the target are lost; secondly, sparse representation SR and sequence information SR have almost the same entropy, but the latter approach offers more details because in the method the HR dictionary is formed by several LR dictionaries; finally, the proposed STTF-based SR method outperforms the other three methods in both metrics.

Figure 10 presents the visual quality of the results obtained by four test methods. We choose the 16th, 21st, 34th,

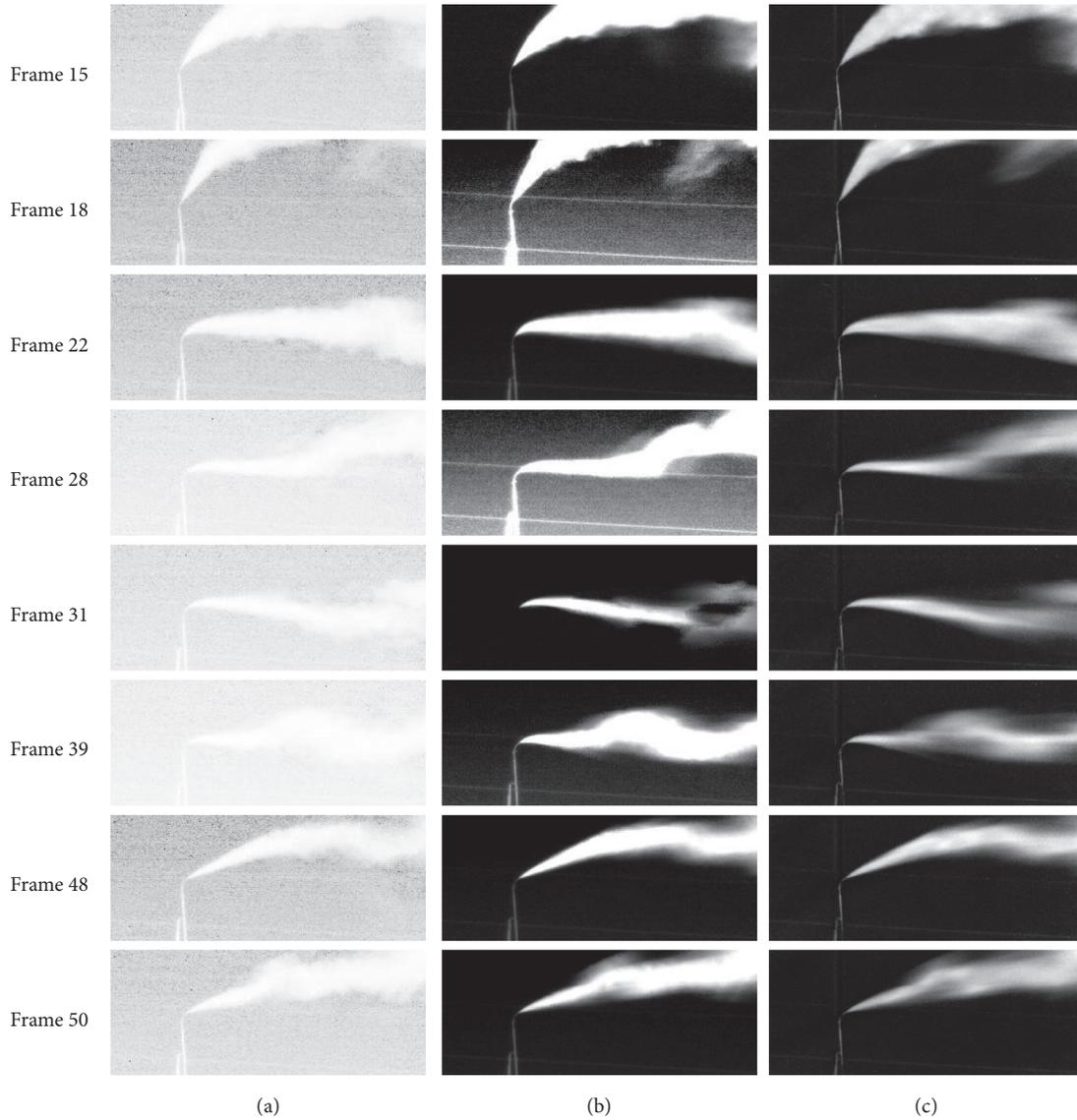


FIGURE 9: The 2D form of the detection results by three detection methods.

TABLE 2: SR quantitative results (entropy and average gradient) of the test methods on key-frames.

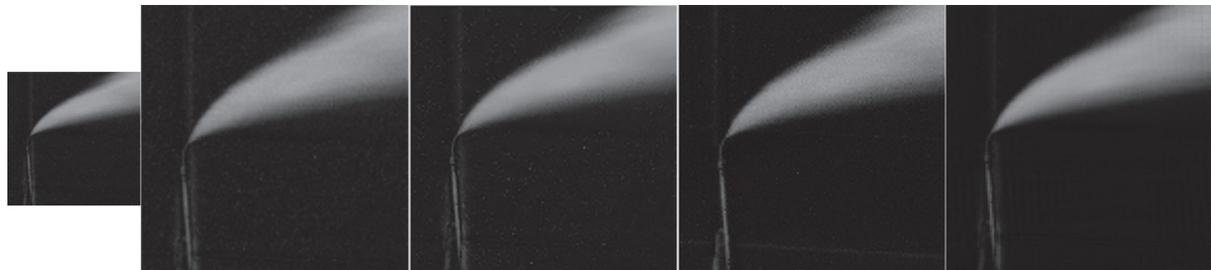
Methods	LR frame		Bicubic interpolation		Sparse representation-based SR [54]		Sequence information-based SR [65]		STTF-based SR	
	Entropy	Average gradient	Entropy	Average gradient	Entropy	Average gradient	Entropy	Average gradient	Entropy	Average gradient
14	5.1603	0.0076	5.3744	0.0052	5.4996	0.0078	5.4259	0.0090	5.6098	0.0121
15	4.7086	0.0077	5.1407	0.0063	5.2398	0.0089	5.2184	0.0103	5.3678	0.0139
16	5.5521	0.0084	5.8013	0.0060	5.8765	0.0085	5.8203	0.0103	5.9772	0.0135
17	5.5293	0.0086	5.7054	0.0056	5.7918	0.0081	5.6125	0.0094	5.8831	0.0129
18	4.2989	0.0072	4.8339	0.0062	4.9794	0.0088	5.0423	0.0106	5.1108	0.0138
19	4.4843	0.0073	5.0045	0.0063	5.1327	0.0089	5.1831	0.0106	5.2644	0.0140
20	5.1442	0.0075	5.4307	0.0060	5.5039	0.0086	5.3987	0.0099	5.6122	0.0137
21	4.8821	0.0071	5.2234	0.0060	5.3264	0.0086	5.2578	0.0100	5.4491	0.0137
22	4.3472	0.0067	4.8929	0.0065	4.9858	0.0090	4.9409	0.0102	5.1261	0.0141
23	4.6127	0.0067	5.0430	0.0061	5.1534	0.0086	5.0806	0.0098	5.2823	0.0135
24	4.4189	0.0064	4.8688	0.0060	4.9850	0.0085	4.8916	0.0096	5.1168	0.0133
25	4.3273	0.0066	4.9091	0.0066	5.0061	0.0091	4.9607	0.0103	5.1494	0.0143

TABLE 2: Continued.

Methods	LR frame		Bicubic interpolation		Sparse representation-based SR [54]		Sequence information-based SR [65]		STTF-based SR	
Frame	Entropy	Average gradient	Entropy	Average gradient	Entropy	Average gradient	Entropy	Average gradient	Entropy	Average gradient
26	4.1394	0.0064	4.7589	0.0066	4.8438	0.0090	4.8078	0.0103	4.9919	0.0141
27	4.1127	0.0065	4.6995	0.0066	4.7878	0.0091	4.7713	0.0104	4.9366	0.0142
28	3.9657	0.0061	4.6576	0.0066	4.7467	0.0091	4.7507	0.0107	4.8902	0.0142
29	4.1885	0.0063	4.7545	0.0064	4.8611	0.0088	4.8481	0.0101	5.0066	0.0137
30	3.9672	0.0063	4.6345	0.0067	4.7253	0.0092	4.7241	0.0106	4.8752	0.0144
31	3.9440	0.0061	4.6135	0.0065	4.7131	0.0090	4.7349	0.0104	4.8654	0.0141
32	3.8661	0.0060	4.5799	0.0064	4.6914	0.0088	4.6971	0.0101	4.8497	0.0138
33	4.0479	0.0060	4.7100	0.0064	4.8126	0.0088	4.8114	0.0100	4.9631	0.0137
34	4.1691	0.0060	4.7824	0.0066	4.8621	0.0088	4.8462	0.0102	5.0072	0.0136
35	4.0933	0.0062	4.7169	0.0067	4.8010	0.0091	4.8245	0.0108	4.9515	0.0143
36	3.9157	0.0063	4.5995	0.0067	4.6881	0.0092	4.6712	0.0103	4.8508	0.0142
37	3.7810	0.0059	4.5028	0.0064	4.6088	0.0089	4.5984	0.0100	4.7666	0.0138
38	3.8814	0.0061	4.5479	0.0065	4.6483	0.0090	4.6395	0.0101	4.8050	0.0140
39	4.3168	0.0060	4.8397	0.0061	4.9406	0.0084	4.9135	0.0099	5.0792	0.0130
40	3.9333	0.0061	4.6597	0.0067	4.7380	0.0091	4.7209	0.0104	4.8906	0.0142
41	4.2009	0.0063	4.7897	0.0066	4.8711	0.0089	4.8346	0.0102	5.0138	0.0138
42	4.1083	0.0063	4.7514	0.0067	4.8362	0.0091	4.8398	0.0103	4.9836	0.0142
43	4.0485	0.0063	4.6827	0.0067	4.7602	0.0091	4.7117	0.0101	4.9109	0.0142
44	4.0521	0.0062	4.6510	0.0063	4.7587	0.0087	4.7273	0.0097	4.9165	0.0136
45	4.3442	0.0060	4.9011	0.0061	5.0079	0.0085	4.9380	0.0097	5.1413	0.0134
46	4.0006	0.0061	4.5913	0.0062	4.7080	0.0087	4.6467	0.0097	4.8587	0.0136
47	4.4749	0.0062	5.0109	0.0060	5.1015	0.0084	5.0007	0.0095	5.2302	0.0131
48	4.0685	0.0065	4.6843	0.0065	4.7859	0.0090	4.7607	0.0101	4.9384	0.0141
Avg.	4.3167	0.0066	4.8671	0.0063	4.9651	0.0088	4.9329	0.0101	5.1049	0.0138



(a)



(b)

FIGURE 10: Continued.

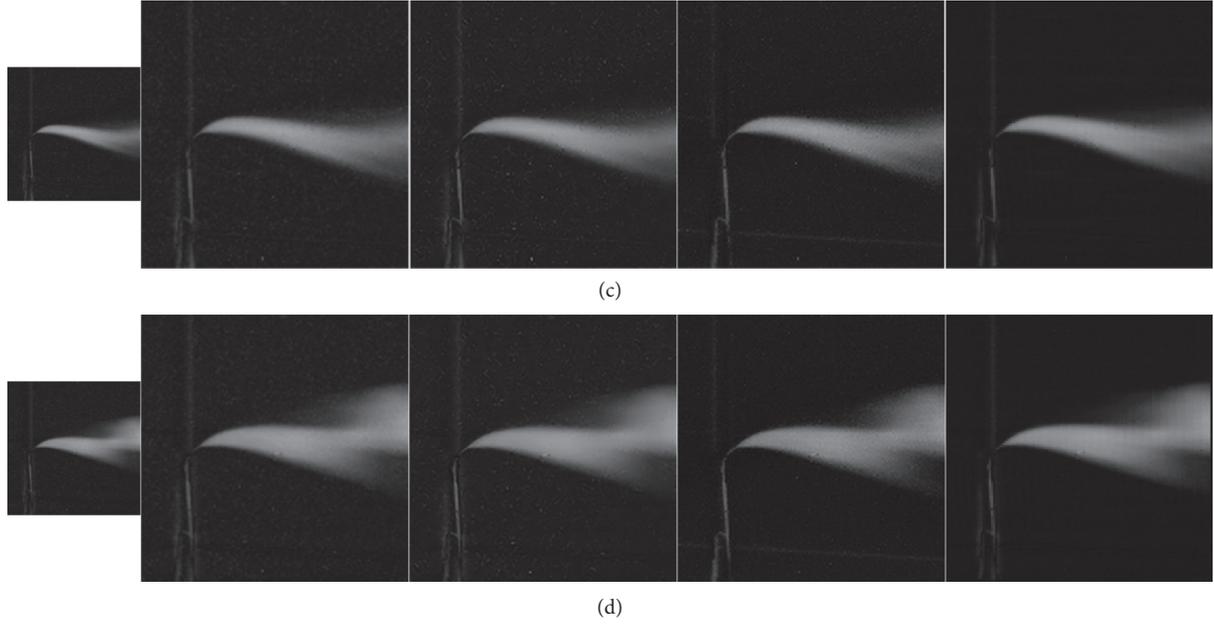


FIGURE 10: The SR results of four test methods, from left to right: original LR frame, bicubic interpolation, sparse representation-based SR method [54], sequence information-based SR method [65], the proposed STTF-based SR method: (a) frame 16; (b) frame 21; (c) frame 34; (d) frame 47.

and 47th frames as a representative. The smaller one with size of 128×128 is the LR 2D-form frame. The bigger ones with size of 256×256 are the SR results of different algorithms. As can be seen from Figure 10, the proposed approach yields clearer outputs with sharper edges and more textures. A drawback is the “checkerboard artifacts,” which may be caused by the deconvolution operations in the method. We desired to fix it in our future work.

4. Conclusions

In this paper, aiming at hyperspectral video, we propose a novel key-frame and target detection method based on cumulative tensor CP factorization, termed as CTCF, and a super-resolution algorithm based on sparse-based tensor Tucker factorization, called STTF. Unlike conventional matrix factorization based methods, CTCF considers hyperspectral video (HSV) as 4D cumulative tensor and approximates new added frames by updating factor matrices. To break the limit of conventional methods and make super-resolution (SR) more practical, STTF exploits the sparsity of HSV frames and factorizes them as a sparse core tensor multiplied by three modes dictionaries. In this way, spatial resolution of LR-HSI is enhanced directly without HR samples. The experimental results systematically prove that the proposed CTCF and STTF methods outperform other state-of-the-art algorithms.

In the future works, we focus on tensor factorization based target tracking methods which are able to extract target region more accurately and clearly. For super-resolution, we aim at exploiting nonlocal similarities in tensor factorization framework, which has been widely used in inverse problems. Besides target tracking and super-resolution, regions of interest (ROI) approaches will be investigated, in order to make

HSV target recognition more efficient and full featured. Inspired by [70] and other related works, we believe that the researches of chemical gas detecting methods will benefit the agricultural application of HSI/HSV. These studies will be of great significance in internet of things (IoT), smart agriculture, pollution monitoring, etc.

Appendix

The optimizations of \mathbf{H} , \mathbf{W} , \mathbf{S} , and \mathbf{Z} in Section 2.3.2 are presented as follows.

- (1) Optimization of \mathbf{H} : when \mathbf{W} , \mathbf{S} , and \mathbf{Z} are fixed, the optimization of \mathbf{H} in (42) is represented as

$$\min_{\mathbf{H}} \|\mathbf{Y} - \mathbf{Z} \times_1 \mathbf{H}^* \times_2 \mathbf{W}^* \times_3 \mathbf{S}\|_F^2 + \alpha \|\mathbf{H} - \mathbf{H}_{\text{pre}}\|_F^2, \quad (\text{A.1})$$

where \mathbf{H}_{pre} denotes the previous estimation of height mode dictionary in last iteration. Using characteristics of n -mode product (see (3)), (A.1) is represented as

$$\min_{\mathbf{H}} \|\mathbf{Y}_{(1)} - \mathbf{D}_1 \mathbf{H} \mathbf{M}_h\|_F^2 + \alpha \|\mathbf{H} - \mathbf{H}_{\text{pre}}\|_F^2, \quad (\text{A.2})$$

where $\mathbf{Y}_{(1)}$ denotes the mode-1 unfolding matrix of \mathbf{Y} and $\mathbf{M}_h = (\mathbf{Z} \times_2 \mathbf{W}^* \times_3 \mathbf{S})_{(1)}$. Equation (A.2) is quadratic and can be solved by computing general Sylvester matrix equation; i.e.,

$$\mathbf{D}_1^T \mathbf{D}_1 \mathbf{H} \mathbf{M}_h \mathbf{M}_h^T + \alpha \mathbf{H} = \mathbf{D}_1^T \mathbf{Y}_{(1)} \mathbf{M}_h^T + \alpha \mathbf{H}_{\text{pre}}. \quad (\text{A.3})$$

The conjugate gradient (CG) method is utilized to solve (A.3). After several iterations, CG will reach the convergence in certain conditions. In our experiments, it has been found that the solution of (A.3) is well approximated after 20 iterations.

(2) Optimization of \mathbf{W} : when \mathbf{H} , \mathbf{S} , and \mathbf{Z} are fixed, the optimization of \mathbf{W} in (42) is expressed by

$$\min_{\mathbf{W}} \|\mathbf{Y} - \mathbf{Z} \times_1 \mathbf{H}^* \times_2 \mathbf{W}^* \times_3 \mathbf{S}\|_F^2 + \alpha \|\mathbf{W} - \mathbf{W}_{\text{pre}}\|_F^2, \quad (\text{A.4})$$

where \mathbf{W}_{pre} denotes the previous estimation of width mode dictionary in last iteration. Similar to the optimization of \mathbf{H} , (A.4) can be transformed into

$$\min_{\mathbf{W}} \|\mathbf{Y}_{(2)} - \mathbf{D}_2 \mathbf{W} \mathbf{M}_w\|_F^2 + \alpha \|\mathbf{W} - \mathbf{W}_{\text{pre}}\|_F^2, \quad (\text{A.5})$$

where $\mathbf{Y}_{(2)}$ denotes the mode-2 unfolding matrix of \mathbf{Y} and $\mathbf{M}_w = (\mathbf{Z} \times_1 \mathbf{H}^* \times_3 \mathbf{S})_{(2)}$. Equation (A.5) is also quadratic and can be solved by computing general Sylvester matrix equation; i.e.,

$$\mathbf{D}_2^T \mathbf{D}_2 \mathbf{W} \mathbf{M}_w \mathbf{M}_w^T + \alpha \mathbf{W} = \mathbf{D}_2^T \mathbf{Y}_{(2)} \mathbf{M}_w^T + \alpha \mathbf{W}_{\text{pre}}. \quad (\text{A.6})$$

Likewise, CG is used to solve (A.6).

(3) Optimization of \mathbf{S} : when \mathbf{H} , \mathbf{W} , and \mathbf{Z} are fixed, the optimization with respect to \mathbf{S} in (42) can be formulated as

$$\min_{\mathbf{S}} \|\mathbf{Y} - \mathbf{Z} \times_1 \mathbf{H}^* \times_2 \mathbf{W}^* \times_3 \mathbf{S}\|_F^2 + \alpha \|\mathbf{S} - \mathbf{S}_{\text{pre}}\|_F^2, \quad (\text{A.7})$$

where \mathbf{S}_{pre} denotes the previous estimation of spectral mode dictionary in last iteration. Same as the processing in the two subsections above, we have

$$\min_{\mathbf{S}} \|\mathbf{Y}_{(3)} - \mathbf{S} \mathbf{M}_s\|_F^2 + \alpha \|\mathbf{S} - \mathbf{S}_{\text{pre}}\|_F^2, \quad (\text{A.8})$$

where $\mathbf{Y}_{(3)}$ denotes the mode-3 unfolding matrix of \mathbf{Y} and $\mathbf{M}_s = (\mathbf{Z} \times_1 \mathbf{H}^* \times_2 \mathbf{W}^*)_{(3)}$. Similarly, (A.8) can be solved by computing general Sylvester matrix equation; i.e.,

$$\mathbf{S} \mathbf{M}_s \mathbf{M}_s^T + \alpha \mathbf{S} = \mathbf{Y}_{(3)} \mathbf{M}_s^T + \alpha \mathbf{S}_{\text{pre}}. \quad (\text{A.9})$$

We apply CG to solve (A.9) and the convergence is achieved in a few iterations.

(4) Optimization of \mathbf{Z} : when \mathbf{H} , \mathbf{W} , and \mathbf{S} are fixed, the optimization of \mathbf{Z} in (42) can be written as

$$\min_{\mathbf{Z}} \|\mathbf{Y} - \mathbf{Z} \times_1 \mathbf{H}^* \times_2 \mathbf{W}^* \times_3 \mathbf{S}\|_F^2 + \lambda \|\mathbf{Z}\|_1 + \alpha \|\mathbf{Z} - \mathbf{Z}_{\text{pre}}\|_F^2, \quad (\text{A.10})$$

where \mathbf{Z}_{pre} denotes the previous estimation of core tensor in last iteration. Equation (A.10) is convex, so we can employ the ADMM to solve the optimization problem. Introducing splitting variables $\mathbf{Z}_1 = \mathbf{Z}$ and $\mathbf{Z}_2 = \mathbf{Z}$, (A.10) can be transformed into the equivalent constrained form:

$$\begin{aligned} & \min_{\mathbf{Z}_1, \mathbf{Z}_2} f_1(\mathbf{Z}_1) + f_2(\mathbf{Z}_2), \\ & \text{s.t. } \mathbf{Z}_1 = \mathbf{Z}, \\ & \quad \mathbf{Z}_2 = \mathbf{Z}, \end{aligned} \quad (\text{A.11})$$

where

$$f_1(\mathbf{Z}_1) = \lambda \|\mathbf{Z}_1\|_1 + \alpha \|\mathbf{Z}_1 - \mathbf{Z}_{1\text{pre}}\|_F^2, \quad (\text{A.12})$$

$$f_2(\mathbf{Z}_2) = \|\mathbf{Y} - \mathbf{Z}_2 \times_1 \mathbf{H}^* \times_2 \mathbf{W}^* \times_3 \mathbf{S}\|_F^2.$$

Equation (A.11) is a typical form of optimization problem that corresponds to the standard ADMM. The augmented Lagrangian function for (A.11) is represented as

$$\begin{aligned} g(\mathbf{Z}_1, \mathbf{Z}_2, \beta) = & \lambda \|\mathbf{Z}_1\|_1 + \alpha \|\mathbf{Z}_1 - \mathbf{Z}_{1\text{pre}}\|_F^2 \\ & + \|\mathbf{Y} - \mathbf{Z}_2 \times_1 \mathbf{H}^* \times_2 \mathbf{W}^* \times_3 \mathbf{S}\|_F^2 \\ & + \mu \|\mathbf{Z}_1 - \mathbf{Z}_2 - \beta\|_F^2, \end{aligned} \quad (\text{A.13})$$

where β denotes the Lagrangian multiplier and μ denotes the penalty parameter. The process of ADMM is formulated as

$$\begin{aligned} \mathbf{Z}_1 & \leftarrow \arg \min_{\mathbf{Z}_1} g(\mathbf{Z}_1, \mathbf{Z}_2, \beta); \\ \mathbf{Z}_2 & \leftarrow \arg \min_{\mathbf{Z}_2} g(\mathbf{Z}_1, \mathbf{Z}_2, \beta); \\ \beta & \leftarrow \beta - (\mathbf{Z}_1 - \mathbf{Z}_2), \end{aligned} \quad (\text{A.14})$$

Here, the optimizations of \mathbf{Z}_1 and \mathbf{Z}_2 are independent because function $g(\cdot)$ is decoupled with respect to these variables. Next, (A.14) will be discussed more detailedly.

(i) Update \mathbf{Z}_1 : based on (A.13), we have

$$\mathbf{Z}_1 \in \arg \min_{\mathbf{Z}_1} \lambda \|\mathbf{Z}_1\|_1 + \alpha \|\mathbf{Z}_1 - \mathbf{Z}_{1\text{pre}}\|_F^2 + \mu \|\mathbf{Z}_1 - \mathbf{Z}_2 - \beta\|_F^2, \quad (\text{A.15})$$

and the closed-form solution of (A.15) is

$$\mathbf{Z}_1 = \text{soft} \left[\frac{\mu(\mathbf{Z}_2 + \beta) + \alpha \mathbf{Z}_{1\text{pre}}}{\mu + \alpha}, \frac{\lambda}{2\mu + 2\alpha} \right], \quad (\text{A.16})$$

where $\text{soft}(m, n) = \text{sign}(m) * \max(|m| - n, 0)$.

(ii) Update \mathbf{Z}_2 : based on (A.13), we have

$$\mathbf{Z}_2 \in \arg \min_{\mathbf{Z}_2} \|\mathbf{Y} - \mathbf{Z}_2 \times_1 \mathbf{H}^* \times_2 \mathbf{W}^* \times_3 \mathbf{S}\|_F^2 + \mu \|\mathbf{Z}_2 - \mathbf{Z}_1 + \beta\|_F^2. \quad (\text{A.17})$$

Based on (6) and (7), (A.17) is equivalent to

$$\arg \min_{\mathbf{z}_2} \|\mathbf{y} - \mathbf{E} \mathbf{z}_2\|_F^2 + \mu \|\mathbf{z}_2 - \mathbf{z}_1 + \beta\|_F^2, \quad (\text{A.18})$$

where the vectors $\mathbf{y} = \text{vec}(\mathbf{Y})$, $\mathbf{z}_2 = \text{vec}(\mathbf{Z}_2)$, $\mathbf{z}_1 = \text{vec}(\mathbf{Z}_1)$, and $\beta = \text{vec}(\beta)$ are the vectorization form of tensors \mathbf{Y} , \mathbf{Z}_2 , \mathbf{Z}_1 , and β , respectively, and matrix $\mathbf{E} = \mathbf{S} \otimes \mathbf{W}^* \otimes \mathbf{H}^*$. Equation (A.18) has the closed-form solution which is denoted by

$$\mathbf{z}_2 = (\mathbf{E}^T \mathbf{E} + \mu \mathbf{I})^{-1} (\mathbf{E}^T \mathbf{y} + \mu \mathbf{z}_1 - \mu \beta). \quad (\text{A.19})$$

However, $\mathbf{E} \in \mathbb{R}^{hwS \times z_1 z_2 z_3}$ is so large that (A.19) is too heavy to be solved. We rewrite the first term of (A.19) as follows:

$$(\mathbf{E}^T \mathbf{E} + \mu \mathbf{I})^{-1} = (\mathbf{P}_3 \otimes \mathbf{P}_2 \otimes \mathbf{P}_1)(\mathbf{Q}_3 \otimes \mathbf{Q}_2 \otimes \mathbf{Q}_1 + \mu \mathbf{I})^{-1} \times (\mathbf{P}_3^T \otimes \mathbf{P}_2^T \otimes \mathbf{P}_1^T), \quad (\text{A.20})$$

where \mathbf{P}_i and \mathbf{Q}_i ($i = [1, 3]$) denote eigenvector matrices and eigenvalue matrices of $\mathbf{H}^{*T} \mathbf{H}^*$, $\mathbf{W}^{*T} \mathbf{W}^*$, and $\mathbf{S}^T \mathbf{S}$, respectively. So, $(\mathbf{Q}_3 \otimes \mathbf{Q}_2 \otimes \mathbf{Q}_1 + \mu \mathbf{I})^{-1}$ is diagonal and can be computed easily. Moreover, the operation of \mathbf{P}_i and of \mathbf{P}_i^T is i -mode products and the multiplication in (A.20) is elementwise. Finally, $\mathbf{E}^T \mathbf{y}$ in the second term of (A.19) can be computed by

$$\mathbf{E}^T \mathbf{y} = \text{vec}(\mathbf{Y} \times_1 \mathbf{H}^{*T} \times_2 \mathbf{W}^{*T} \times_3 \mathbf{S}^T). \quad (\text{A.21})$$

(iii) Update β : based on (A.14), β is updated by

$$\beta \leftarrow \beta - (\mathbf{Z}_1 - \mathbf{Z}_2). \quad (\text{A.22})$$

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare no conflicts of interest.

Acknowledgments

The authors would like to thank Professor Gu from Heilongjiang Province Key Laboratory of Space-Air-Ground Integrated Intelligent Remote Sensing for his selfless help. This work was supported by the National Natural Science Foundation of China (Grant no. 61671184) and the National Natural Science Foundation of Key International Cooperation of China (Grant no. 61720106002).

References

- [1] H. Fan, C. Li, Y. Guo, G. Kuang, and J. Ma, "Spatial-spectral total variation regularized low-rank tensor decomposition for hyperspectral image denoising," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 56, no. 10, pp. 6196–6213, 2018.
- [2] X. Zheng, Y. Yuan, and X. Lu, "Hyperspectral image denoising by fusing the selected related bands," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 57, no. 5, pp. 2596–2609, 2019.
- [3] D. Hong, N. Yokoya, J. Chanussot, and X. X. Zhu, "CoSpace: common subspace learning from hyperspectral-multispectral correspondences," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 57, no. 7, pp. 4349–4359, 2019.
- [4] D. Hong, N. Yokoya, J. Chanussot, J. Xu, and X. X. Zhu, "Learning to propagate labels on graphs: an iterative multitask regression framework for semi-supervised hyperspectral dimensionality reduction," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 158, pp. 35–49, 2019.
- [5] T. Liu, Y. Gu, X. Jia, J. A. Benediktsson, and J. Chanussot, "Class-specific sparse multiple kernel learning for spectral-spatial hyperspectral image classification," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 54, no. 12, pp. 7351–7365, 2016.
- [6] Y. Gu, T. Liu, X. Jia, J. A. Benediktsson, and J. Chanussot, "Nonlinear multiple kernel learning with multiple-structure-element extended morphological profiles for hyperspectral image classification," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 54, no. 6, pp. 3235–3247, 2016.
- [7] T. Liu, Y. Gu, J. Chanussot, and M. Dalla Mura, "Multimorphological superpixel model for hyperspectral image classification," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 55, no. 12, pp. 6950–6963, 2017.
- [8] T. Liu, X. Zhang, and Y. Gu, "Unsupervised cross-temporal classification of hyperspectral images with multiple geodesic flow kernel learning," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 57, no. 12, pp. 9688–9701, 2019.
- [9] Y. Gu, T. Liu, and J. Li, "Superpixel tensor model for spatial-spectral classification of remote sensing images," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 57, no. 7, pp. 4705–4719, 2019.
- [10] D. Hong, N. Yokoya, N. Ge, J. Chanussot, and X. X. Zhu, "Learnable manifold alignment (LeMA): a semi-supervised cross-modality learning framework for land cover and land use classification," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 147, pp. 193–205, 2019.
- [11] M. Song, X. Shang, Y. Wang, C. Yu, and C.-I. Chang, "Class information-based band selection for hyperspectral image classification," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 57, no. 11, pp. 8394–8416, 2019.
- [12] Y. Zhang, W. Ke, B. Du, and X. Hu, "Independent encoding joint sparse representation and multitask learning for hyperspectral target detection," *IEEE Geoscience and Remote Sensing Letters*, vol. 14, no. 11, pp. 1933–1937, 2017.
- [13] N. M. Nasrabadi, "Hyperspectral target detection: an overview of current and future challenges," *IEEE Signal Processing Magazine*, vol. 31, no. 1, pp. 34–44, 2014.
- [14] Y. Wang, L. Wang, C. Yu et al., "Constrained-target band selection for multiple-target detection," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 57, no. 8, pp. 6079–6103, 2019.
- [15] R. Dian, S. Li, and L. Fang, "Learning a low tensor-train rank representation for hyperspectral image super-resolution," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 30, no. 9, pp. 2672–2683, 2019.
- [16] L. Fang, H. Zhuo, and S. Li, "Super-resolution of hyperspectral image via superpixel-based sparse representation," *Neurocomputing*, vol. 273, no. 17, pp. 171–177, 2018.
- [17] R. Dian, L. Fang, and S. Li, "Hyperspectral image super-resolution via non-local sparse tensor factorization," in *Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3862–3871, Honolulu, HI, USA, July 2017.
- [18] Y. Wang, L. Wang, H. Xie, and C.-I. Chang, "Fusion of various band selection methods for hyperspectral imagery," *Remote Sensing*, vol. 11, no. 18, p. 2125, 2019.
- [19] Z. Huang, L. Fang, and S. Li, "Subpixel-pixel-superpixel guided fusion for hyperspectral anomaly detection," *IEEE Transactions on Geoscience and Remote Sensing*, pp. 1–10, 2020.
- [20] X. Zhang, G. Wen, and W. Dai, "A tensor decomposition-based anomaly detection algorithm for hyperspectral image," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 54, no. 10, pp. 5801–5820, 2016.
- [21] S. Li, K. Zhang, Q. Hao, P. Duan, and X. Kang, "Hyperspectral anomaly detection with multiscale attribute and edge-preserving filters," *IEEE Geoscience and Remote Sensing Letters*, vol. 15, no. 10, pp. 1605–1609, 2018.

- [22] Y. Wang, L.-C. Lee, B. Xue et al., "A posteriori hyperspectral anomaly detection for unlabeled classification," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 56, no. 6, pp. 3091–3106, 2018.
- [23] I. S. Reed and X. Yu, "Adaptive multiple-band CFAR detection of an optical pattern with unknown spectral distribution," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 38, no. 10, pp. 1760–1770, 1990.
- [24] D. Meng, X. Wang, M. Huang, L. Wan, and B. Zhang, "Robust weighted subspace fitting for DOA estimation via block sparse recovery," *IEEE Communications Letters*, vol. 24, no. 3, pp. 563–567, 2020.
- [25] L. L. Scharf and B. Friedlander, "Matched subspace detectors," *IEEE Transactions on Signal Processing*, vol. 42, no. 8, pp. 2146–2157, 1994.
- [26] D. Hong, N. Yokoya, J. Chanussot, and X. X. Zhu, "An augmented linear mixing model to address spectral variability for hyperspectral unmixing," *IEEE Transactions on Image Processing*, vol. 28, no. 4, pp. 1923–1938, 2019.
- [27] X. Li, X. Jia, L. Wang, and K. Zhao, "On spectral unmixing resolution using extended support vector machines," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 53, no. 9, pp. 4985–4996, 2015.
- [28] M.-D. Iordache, J. M. Bioucas-Dias, and A. Plaza, "Sparse unmixing of hyperspectral data," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 49, no. 6, pp. 2014–2039, 2011.
- [29] N. Akhtar and A. Mian, "RCMF: robust constrained matrix factorization for hyperspectral unmixing," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 55, no. 6, pp. 3354–3366, 2017.
- [30] Q. Wei, N. Dobigeon, and J. Tourneret, "Bayesian fusion of hyperspectral and multispectral images," in *Proceedings of the 2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 3176–3180, Florence, Italy, May 2014.
- [31] V. P. Shah, N. H. Younan, and R. L. King, "An efficient pansharpening method via a combined adaptive PCA approach and contourlets," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 46, no. 5, pp. 1323–1335, 2008.
- [32] R. Dian, S. Li, A. Guo, and L. Fang, "Deep hyperspectral image sharpening," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 29, no. 11, pp. 5345–5355, 2018.
- [33] W. Dong, F. Fu, G. Shi et al., "Hyperspectral image super-resolution via non-negative structured sparse representation," *IEEE Transactions on Image Processing*, vol. 25, no. 5, pp. 2337–2352, 2016.
- [34] B. Huang, H. Song, H. Cui, J. Peng, and Z. Xu, "Spatial and spectral image fusion using sparse matrix factorization," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 52, no. 3, pp. 1693–1704, 2014.
- [35] N. Akhtar, F. Shafait, and A. Mian, "Bayesian sparse representation for hyperspectral image super resolution," in *Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3631–3640, Boston, MA, USA, June 2015.
- [36] X. Fu, K. Huang, B. Yang, W.-K. Ma, and N. D. Sidiropoulos, "Robust volume minimization-based matrix factorization for remote sensing and document clustering," *IEEE Transactions on Signal Processing*, vol. 64, no. 23, pp. 6254–6268, 2016.
- [37] X. Fu, K. Huang, and N. D. Sidiropoulos, "On identifiability of nonnegative matrix factorization," *IEEE Signal Processing Letters*, vol. 25, no. 3, pp. 328–332, 2018.
- [38] K. Huang, N. D. Sidiropoulos, and A. Swami, "Non-negative matrix factorization revisited: uniqueness and algorithm for symmetric decomposition," *IEEE Transactions on Signal Processing*, vol. 62, no. 1, pp. 211–224, 2014.
- [39] W.-K. Ma, J. M. Bioucas-Dias, T.-H. Chan et al., "A signal processing perspective on hyperspectral unmixing: insights from remote sensing," *IEEE Signal Processing Magazine*, vol. 31, no. 1, pp. 67–81, 2014.
- [40] N. Yokoya, T. Yairi, and A. Iwasaki, "Coupled nonnegative matrix factorization unmixing for hyperspectral and multispectral data fusion," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 50, no. 2, pp. 528–537, 2012.
- [41] C. Lanaras, E. Baltsavias, and K. Schindler, "Hyperspectral super-resolution by coupled spectral unmixing," in *Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV)*, pp. 3586–3594, Santiago, Chile, December 2015.
- [42] K. Zhang, M. Wang, and S. Yang, "Multispectral and hyperspectral image fusion based on group spectral embedding and low-rank factorization," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 55, no. 3, pp. 1363–1371, 2017.
- [43] Y. Xu, Z. Wu, J. Chanussot, and Z. Wei, "Nonlocal patch tensor sparse representation for hyperspectral image super-resolution," *IEEE Transactions on Image Processing*, vol. 28, no. 6, pp. 3034–3047, 2019.
- [44] Y. Xu, Z. Wu, J. Chanussot, and Z. Wei, "Hyperspectral images super-resolution via learning high-order coupled tensor ring representation," *IEEE Transactions on Neural Networks and Learning Systems*, pp. 1–14, 2020.
- [45] R. Dian and S. Li, "Hyperspectral image super-resolution via subspace-based low tensor multi-rank regularization," *IEEE Transactions on Image Processing*, vol. 28, no. 10, pp. 5135–5146, 2019.
- [46] T. G. Kolda and B. W. Bader, "Tensor decompositions and applications," *Siam Review*, vol. 51, no. 3, pp. 455–500, 2009.
- [47] H. A. L. Kiers, "Towards a standardized notation and terminology in multiway analysis," *Journal of Chemometrics*, vol. 14, no. 3, pp. 105–122, 2000.
- [48] J. B. Kruskal, "Three-way arrays: rank and uniqueness of trilinear decompositions, with application to arithmetic complexity and statistics," *Linear Algebra and its Applications*, vol. 18, no. 2, pp. 95–138, 1977.
- [49] A. Smilde, R. Bro, and P. Geladi, *Multi-Way Analysis: Applications in the Chemical Sciences*, Wiley, West Sussex, England, UK, 2004.
- [50] L. R. Tucker, "Some mathematical notes on three-mode factor analysis," *Psychometrika*, vol. 31, no. 3, pp. 279–311, 1966.
- [51] G. H. Golub and C. F. Van Loan, *Matrix Computations*, Johns Hopkins University Press, Baltimore, MD, USA, 1996.
- [52] L. Loncan, L. B. de Almeida, J. M. Bioucas-Dias et al., "Hyperspectral pansharpening: a review," *IEEE Geoscience and Remote Sensing Magazine*, vol. 3, no. 3, pp. 27–46, 2015.
- [53] L. Zhuang and J. M. Bioucas-Dias, "Fast hyperspectral image denoising and inpainting based on low-rank and sparse representations," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 11, no. 3, pp. 730–742, 2018.
- [54] J. Yang, J. Wright, T. S. Huang, and Y. Ma, "Image super-resolution via sparse representation," *IEEE Transactions on Image Processing*, vol. 19, no. 11, pp. 2861–2873, 2010.
- [55] T. Lu, S. Li, L. Fang, Y. Ma, and J. A. Benediktsson, "Spectral-spatial adaptive sparse representation for hyperspectral image

- denoising,” *IEEE Transactions on Geoscience and Remote Sensing*, vol. 54, no. 1, pp. 373–385, 2016.
- [56] R. Arablouei and F. de Hoog, “Hyperspectral image recovery via hybrid regularization,” *IEEE Transactions on Image Processing*, vol. 25, no. 12, pp. 5649–5663, 2016.
- [57] L. Zhang, W. Wei, C. Tian, F. Li, and Y. Zhang, “Exploring structured sparsity by a reweighted Laplace prior for hyperspectral compressive sensing,” *IEEE Transactions on Image Processing*, vol. 25, no. 10, pp. 4974–4988, 2016.
- [58] L. Fang, C. Wang, S. Li, and J. A. Benediktsson, “Hyperspectral image classification via multiple-feature-based adaptive sparse representation,” *IEEE Transactions on Instrumentation and Measurement*, vol. 66, no. 7, pp. 1646–1657, 2017.
- [59] H. Attouch, J. Bolte, and B. F. Svaiter, “Convergence of descent methods for semi-algebraic and tame problems: proximal algorithms, forward-backward splitting, and regularized gauss-seidel methods,” *Mathematical Programming*, vol. 137, no. 1-2, pp. 91–129, 2013.
- [60] H. Attouch, J. Bolte, P. Redont, and A. Soubeyran, “Proximal alternating minimization and projection methods for non-convex problems: an approach based on the Kurdyka-Łojasiewicz inequality,” *Mathematics of Operations Research*, vol. 35, no. 2, pp. 438–457, 2010.
- [61] O. Axelsson, *Iterative Solution Methods*, Cambridge Univ. Press, Cambridge, UK, 1996.
- [62] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, “Distributed optimization and statistical learning via the alternating direction method of multipliers,” *Foundations and Trends in Machine Learning*, vol. 3, no. 1, pp. 1–122, 2010.
- [63] L. N. Smith and M. Elad, “Improving dictionary learning: multiple dictionary updates and coefficient reuse,” *IEEE Signal Processing Letters*, vol. 20, no. 1, pp. 79–82, 2013.
- [64] J. M. Bioucas-Dias, “A variable splitting augmented Lagrangian approach to linear spectral unmixing,” in *Proceedings of the 2009 First Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing*, pp. 1–4, Grenoble, France, August 2009.
- [65] R. Zhou, G. Wang, D. Zhao, Y. Zou, and T. Zhang, “Super-resolution of low-quality images based on compressed sensing and sequence information,” in *Proceedings of the 2019 IEEE 90th Vehicular Technology Conference (VTC2019-Fall)*, pp. 1–5, Honolulu, HI, USA, September 2019.
- [66] W. Li and Q. Du, “Collaborative representation for hyperspectral anomaly detection,” *IEEE Transactions on Geoscience and Remote Sensing*, vol. 53, no. 3, pp. 1463–1474, 2015.
- [67] Y. Zhang, B. Du, L. Zhang, and S. Wang, “A low-rank and sparse matrix decomposition-based Mahalanobis distance method for hyperspectral anomaly detection,” *IEEE Transactions on Geoscience and Remote Sensing*, vol. 54, no. 3, pp. 1376–1389, 2016.
- [68] L. Wang, C.-I. Chang, L.-C. Lee et al., “Band subset selection for anomaly detection in hyperspectral imagery,” *IEEE Transactions on Geoscience and Remote Sensing*, vol. 55, no. 9, pp. 4887–4898, 2017.
- [69] X. Wang, L. Wan, M. Huang, C. Shen, Z. Han, and T. Zhu, “Low-complexity channel estimation for circular and non-circular signals in virtual MIMO vehicle communication systems,” *IEEE Transactions on Vehicular Technology*, vol. 69, no. 4, pp. 3916–3928, 2020.
- [70] W. Lu, X. Xu, G. Huang et al., “Energy efficiency optimization in SWIPT enabled WSNs for smart agriculture,” *IEEE Transactions on Industrial Informatics*, p. 1, 2020.