FastVGBS: A Fast Version of the Volume-Gradient-Based Band Selection Method for Hyperspectral Imagery

Luyan Ji, Liangliang Zhu, Lei Wang, Yanxin Xi, Kai Yu, and Xiurui Geng

***Abstract*— Recently, the volume-gradient-based band selection (VGBS) method has attracted more and more attention in the field of band selection. It is a ranking-based unsupervised algorithm which applies the sequential backward selection strategy to successively remove the most abundant band. The key finding of VGBS is that the band redundancy corresponds to the volume gradient matrix with respect to hyperspectral images. However, we have found that VGBS requires to update the gradient matrix after each band removal, which includes the calculation of the matrix inverse, determinant, and multiplication, and thus is time-consuming when the number of bands is large. In this letter, we first find that the norm of the row of the gradient matrix has a one-to-one correspondence to the diagonal element of the covariance matrix of the image. Further, we develop a recursive formula to calculate the inverse of the covariance matrix. The experimental results show the effectiveness of the method, i.e., we can reduce the computational complexity of VGBS with an order of magnitude.**

***Index Terms*— Band selection, hyperspectral, volume-gradient band selection.**

I. Introduction

IN HYPERSPECTRAL remote sensing, images usually consist of hundreds of bands, and the spectral resolution could be as high as less than 1 nm, which provides a great opportunity for target detection/identification [1]. Still, high dimensionality could bring about problems such as high correlation between bands, high computational cost, high storage requirement, and so on [2]. Therefore, dimensionality reduction becomes an important preprocessing step in hyperspectral applications, which includes feature extraction and band selection. Since band selection can retain the physical meaning of bands, it has become more and more preferable in the recent years [1], [3].

In the field of band selection, it can be divided into two categories: supervised and unsupervised band selection methods. The former is based on criteria such as class separability and requires training samples, which is often hard to obtain in practical applications. On the contrary, unsupervised ones do not need prior knowledge, and two major groups can be summarized: ranking-based and clustering-based. The first type uses a criterion to measure the band importance, such as information divergence band selection (IDBS) [4], maximum ellipsoid volume (MEV) [5], orthogonal projection-based band selection [1], Boltzmann entropy-based band selection [6], and so on. Moreover, the second type tries to select bands that lie close to the centers of the clusters, for instance, exemplar component analysis (ECA) [7], enhanced fast density peak-based clustering (E-FDPC) [8], automatic band selection (ABS) [9], and so on.

Recently, the volume-gradient-based band selection (VGBS) [3] has received more and more attention since its publication. VGBS is a ranking-based unsupervised band selection method and adopts the maximum volume criterion used by MEV. By introducing the proportional relation between the subvolume and the volume gradient of a simplex with respect to the left bands, VGBS iteratively removes the band with the largest volume gradient. It can achieve a good classification performance when used as a preprocessing step [1]. Still, VGBS needs to recalculate the gradient matrix when removing a band at a time, which has a computational complexity of $O(L^3)$ (where L is the number of bands). Therefore, VGBS has no advantage in computing speed compared with other methods [1]. Especially, when L is large, the computing time of VGBS will be high and not suitable for real-time processing.

In this letter, to reduce the computational complexity of VGBS, we first present the relation between the gradient matrix and the covariance matrix of the data set, which can be utilized to reduce the complexity of computing gradient matrix. Next, the equation between the inverse matrix of the covariance matrix and the subcovariance by removing a band is proposed, which can further reduce the complexity. The proposed method is mathematically equivalent to the original VGBS, and the real hyperspectral data are used for computing time evaluation, demonstrating that the method is much more computationally efficient.

II. Method

A. VGBS

VGBS is an unsupervised method, which successively removes the most redundant band based on the gradient of volume [3]. It is found that the subvolume is proportional to

Manuscript received June 19, 2019; revised October 14, 2019; accepted March 9, 2020. Date of publication March 20, 2020; date of current version February 25, 2021. This work was supported in part by the National Key Research and Development Program of China under Grant 2016YFA0600103 and in part by the High Resolution Satellite 5 Application Common Key Technology under Grant 30-Y20A28-9004-15/17. *(Corresponding author: Xiurui Geng.)*

The authors are with the Key Laboratory of Technology in Geo-Spatial Information Processing and Application System, Chinese Academy of Sciences, Beijing 100190, China, also with the Aerospace Information Research Institute, Chinese Academy of Sciences, Beijing 100094, China, and also with the School of Electronic, Electrical and Communication Engineering, University of the Chinese Academy of Sciences, Beijing 100049, China (e-mail: gengxr@sina.com.cn).

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

Digital Object Identifier 10.1109/LGRS.2020.2980108

the corresponding gradient of the whole volume with respect to the left band as $V_{m-1} = (m-1)!(dV_m/dh)$, where V_m and V_{m-1} are the volumes of the simplex determined by m and $m-1$ vertexes, respectively, and h is the height from the left band to this subsimplex. Suppose the hyperspectral data set $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_L]^{\mathrm{T}}$ is centralized, where $\mathbf{x}_i$ is an N-dimensional column vector for the ith band, L is the number of bands, and N is the number of pixels. Then, the volume of the parallelotope constructed by the bands is $V = (\det(\mathbf{X}\mathbf{X}^{\mathrm{T}}))^{1/2}$. Then, the derivative of $\det(\mathbf{X}\mathbf{X}^{\mathrm{T}})$ to $\mathbf{X}$, denoted $\mathbf{D}$, is

$$\mathbf{D} = \frac{\partial \det(\mathbf{X}\mathbf{X}^{\mathrm{T}})}{\partial \mathbf{X}} = 2\det(\mathbf{X}\mathbf{X}^{\mathrm{T}})(\mathbf{X}\mathbf{X}^{\mathrm{T}})^{-1}\mathbf{X}. \tag{1}$$

Let $\mathbf{D} = [\mathbf{d}_1, \mathbf{d}_2, \ldots, \mathbf{d}_L]^{\mathrm{T}}$, $\mathbf{d}_i = [d_{i1}, d_{i2}, \ldots, d_{iN}]^{\mathrm{T}}$, and the band selection strategy of VGBS is to successively remove the band with the maximum gradient, that is

$$i_{\mathrm{del}} = \arg \max_{i=1,\ldots,k} (\|\mathbf{d}_i\|). \tag{2}$$

To reduce the computational complexity, VGBS further applies the singular value decomposition (SVD) to $\mathbf{X}$ to derive the data after dimensionality reduction, $\mathbf{Y} = \mathbf{U}\Sigma$, where $\mathbf{U}$ and Σ can be computed by performing SVD to $\mathbf{X}\mathbf{X}^{\mathrm{T}}$ as $\mathbf{X}\mathbf{X}^{\mathrm{T}} = \mathbf{U}\Sigma^2\mathbf{U}^{\mathrm{T}}$. Then, the gradient matrix of the determinant to $\mathbf{Y}$ is

$$\mathbf{D} = 2\det(\mathbf{Y}\mathbf{Y}^{\mathrm{T}})((\mathbf{Y}\mathbf{Y}^{\mathrm{T}})^{-1})^{\mathrm{T}}\mathbf{Y} \tag{3}$$

where $\mathbf{Y}$ is an $L \times L$ matrix. Still, one has to update the gradient matrix $\mathbf{D}$ each time when removing a band, which includes several time-costing parts, such as the computation of the matrix multiplication, inverse, and determinant, as shown in (3). Hence, when the selected number of bands is small and/or the number of bands L is large, VGBS still requires long computing time.

B. FastVGBS

To further reduce the computational complexity of VGBS, we propose a fast version for VGBS, named the fast VGBS (FastVGBS) from two aspects: 1) simplifying the calculation for $\|\mathbf{d}_i\|$ in (2) and 2) accelerating the computation of inverse matrix.

1) Simplifying the Calculation of $\|\mathbf{d}_i\|$*:* The key process of VGBS is to compute $\|\mathbf{d}_i\|$ in (2), which is by calculating the norm of each row of $\mathbf{D}$. In fact, it can also be computed from $\mathbf{D}\mathbf{D}^{\mathrm{T}}$ because

$$\begin{aligned}\mathbf{D}\mathbf{D}^{\mathrm{T}} &= \begin{bmatrix}\mathbf{d}_1\\ \mathbf{d}_2\\ \vdots\\ \mathbf{d}_L\end{bmatrix}\left[\mathbf{d}_1^{\mathrm{T}}, \mathbf{d}_2^{\mathrm{T}}, \ldots, \mathbf{d}_L^{\mathrm{T}}\right]\\ &= \begin{bmatrix}\mathbf{d}_1\mathbf{d}_1^{\mathrm{T}} & \mathbf{d}_1\mathbf{d}_2^{\mathrm{T}} & \cdots & \mathbf{d}_1\mathbf{d}_L^{\mathrm{T}}\\ \mathbf{d}_2\mathbf{d}_1^{\mathrm{T}} & \mathbf{d}_2\mathbf{d}_2^{\mathrm{T}} & \cdots & \mathbf{d}_2\mathbf{d}_L^{\mathrm{T}}\\ \vdots & \vdots & \ddots & \vdots\\ \mathbf{d}_L\mathbf{d}_1^{\mathrm{T}} & \mathbf{d}_L\mathbf{d}_2^{\mathrm{T}} & \cdots & \mathbf{d}_L\mathbf{d}_L^{\mathrm{T}}\end{bmatrix}.\end{aligned} \tag{4}$$

Hence, we can have

$$\begin{aligned}[\|\mathbf{d}_1\|^2, \|\mathbf{d}_2\|^2, \ldots, \|\mathbf{d}_L\|^2] &= [\mathbf{d}_1\mathbf{d}_1^{\mathrm{T}}, \mathbf{d}_2\mathbf{d}_2^{\mathrm{T}}, \ldots, \mathbf{d}_L\mathbf{d}_L^{\mathrm{T}}]\\ &= \mathrm{diag}(\mathbf{D}\mathbf{D}^{\mathrm{T}})\end{aligned} \tag{5}$$

which indicates that the norm of each row of $\mathbf{D}$ can be directly obtained from the diagonal elements of $\mathbf{D}\mathbf{D}^{\mathrm{T}}$. Further, let $\mathbf{K} = \mathbf{X}\mathbf{X}^{\mathrm{T}}$, which corresponds the covariance matrix of $\mathbf{X}$ (it should be noted here that the actual covariance matrix has a scalar, $1/N$. Still, we call $\mathbf{K}$ the covariance matrix without causing ambiguity), then substitute (1) into $\mathbf{D}\mathbf{D}^{\mathrm{T}}$ and we can have

$$\begin{aligned}\mathbf{D}\mathbf{D}^{\mathrm{T}} &= 2\det(\mathbf{K})\mathbf{K}^{-1}\mathbf{X}(2\det(\mathbf{K})\mathbf{K}^{-1}\mathbf{X})^{\mathrm{T}}\\ &= 4(\det(\mathbf{K}))^2\mathbf{K}^{-1}\mathbf{K}\mathbf{K}^{-1}\\ &= 4(\det(\mathbf{K}))^2\mathbf{K}^{-1} = c\cdot\mathbf{K}^{-1}\end{aligned} \tag{6}$$

where the parameter $c = 4(\det(\mathbf{K}))^2$. Substitute (6) into (5) yielding

$$[\|\mathbf{d}_1\|^2, \|\mathbf{d}_2\|^2, \ldots, \|\mathbf{d}_L\|^2] = c\cdot\mathrm{diag}(\mathbf{K}^{-1}). \tag{7}$$

It is interesting to find that the norm of each row of $\mathbf{D}$, $\|\mathbf{d}_i\|^2$ actually equals to the diagonal elements of the inverse matrix of the covariance matrix. It indicates that the diagonal elements of $\mathbf{K}^{-1}$ are the real meaningful metrics for VGBS.

2) Accelerating the Calculation of $\mathbf{K}^{-1}$*:* According to (7), we have converted the calculation of $\|\mathbf{d}_i\|$ into the calculation of $\mathbf{K}^{-1}$. As a result, we have to recalculate $\mathbf{K}^{-1}$ each time when removing a band, which is still time-consuming. Here, we will accelerate the computation of $\mathbf{K}^{-1}$. Suppose $\mathbf{K}_k$ is the covariance matrix of a data set with k bands, and $\mathbf{K}_k^{-1}$ is the corresponding inverse matrix, both of which are already known. Without the loss of generality, we assume the band to be removed is the last one. Then, $\mathbf{K}_k$ and $\mathbf{K}_k^{-1}$ can be partitioned as follows:

$$\mathbf{K}_k = \begin{bmatrix}\mathbf{K}_{k-1} & \mathbf{u}\\ \mathbf{u}^{\mathrm{T}} & a\end{bmatrix} \tag{8}$$

$$\mathbf{K}_k^{-1} = \begin{bmatrix}\hat{\mathbf{K}}_{k-1} & \hat{\mathbf{u}}\\ \hat{\mathbf{u}}^{\mathrm{T}} & \hat{a}\end{bmatrix} \tag{9}$$

where $\mathbf{K}_k$ and $\hat{\mathbf{K}}_{k-1}$ are $(k-1)\times(k-1)$ matrices, representing the remaining matrices after band removal, $\mathbf{u}$ and $\hat{\mathbf{u}}$ are $(k-1)$-dimensional column vectors, and a and $\hat{a}$ are scalars. Now, the problem becomes how to derive $\mathbf{K}_{k-1}^{-1}$ when we know $\mathbf{K}_k^{-1}$. Based on the blockwise inversion equation [10], we can have the inversion matrix of $\mathbf{K}_k$ in (8).

Combining (9) and (10), as shown at the bottom of the next page, we can obtain

$$\hat{a} = \left(a - \mathbf{u}^{\mathrm{T}}\mathbf{K}_{k-1}^{-1}\mathbf{u}\right)^{-1} \tag{11}$$

$$\hat{\mathbf{u}} = -\mathbf{K}_{k-1}^{-1}\mathbf{u}\left(a - \mathbf{u}^{\mathrm{T}}\mathbf{K}_{k-1}^{-1}\mathbf{u}\right)^{-1} \tag{12}$$

$$\hat{\mathbf{K}}_{k-1} = \mathbf{K}_{k-1}^{-1} + \mathbf{K}_{k-1}^{-1}\mathbf{u}\left(a - \mathbf{u}^{\mathrm{T}}\mathbf{K}_{k-1}^{-1}\mathbf{u}\right)^{-1}\mathbf{u}^{\mathrm{T}}\mathbf{K}_{k-1}^{-1}. \tag{13}$$

Substitute (11) into (12) yielding

$$\mathbf{K}_{k-1}^{-1}\mathbf{u} = -\hat{\mathbf{u}}/\hat{a} \tag{14}$$

then substitute the above equation into (13), and we can have

$$\mathbf{K}_{k-1}^{-1} = \hat{\mathbf{K}}_{k-1} - \hat{\mathbf{u}}\hat{\mathbf{u}}^{\mathrm{T}}/\hat{a}. \tag{15}$$

Therefore, we obtain a recursive formula for calculating the inversion of the covariance matrix. It successfully simplifies the calculation of inverse matrix into the vector/scalar multiplication and matrix subtraction, which can greatly reduce the computational complexity. The pseudocode of FastVGBS is presented in Algorithm 1.

Algorithm 1 Fast Version of VGBS, FastVGBS

Input: observations $\mathbf{X}_{L\times N}$, the number of selected bands n
Output: the selected n band indices, $indices$

1: Initialization: (a) Compute $\mathbf{K} = \mathbf{X}\mathbf{X}^{\mathrm{T}}$, and the inversion matrix, $\mathbf{K}^{-1}$. (b) Let $k = L$, and $indices = [1, 2, \ldots, L]$.
- - - - - - - - - - - -($band\ selection$)- - - - - - - - - - - -
2: while $k > n$ do
3: Find the index of the band to be removed similar to (2), $i_{\mathrm{del}} = \arg\max_{i=1,\ldots,k} \mathrm{diag}\left(\mathbf{K}_k^{-1}\right)$.
4: Remove the $i_{\mathrm{del}}^{\mathrm{th}}$ band number from $indices$.
5: Update $\mathbf{K}_k^{-1}$ according to (15).
6: $k \leftarrow k - 1$
7: endwhile

III. Computational Complexity Analysis

In this section, we mainly discuss the computational complexity of FastVGBS compared with the original VGBS (measured in the number of floating-point operations, flops). Since the computational complexity of the initialization step is $O\left(NL^2 + L^3\right)$ for both algorithms, only the complexity of the band selection steps is considered here, which is shown in Table I.

From the band selection process of FastVGBS in Algorithm 1, we can see that the only time consuming step is Step 5 which is to compute the kth $\mathbf{K}^{-1}$ $(1 \le k \le L)$. According to (15), its computational complexity is $O(k^2)$. When n bands are selected, the total computational complexity is $O(\sum_{k=n+1}^{L-1} k^2)$.

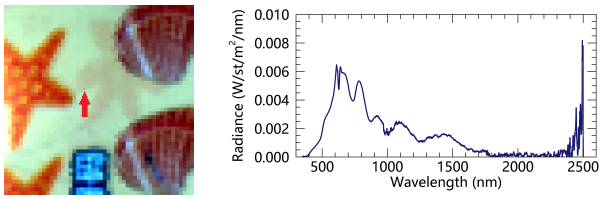
Therefore, by 1) introducing the relation between gradient matrix $\mathbf{D}$ and the inverse matrix of covariance matrix $\mathbf{K}^{-1}$ and 2) accelerating the calculation of $\mathbf{K}^{-1}$, FastVGBS successfully transfers a $O(k^3)$ step of VGBS into a $O(k^2)$ one. As a result, the total computational complexity of the band selection processing is reduced by an order of magnitude compared to that of VGBS.

IV. Experiments

In this section, we mainly compare the computational efficiency of VGBS and FastVGBS since the results of both algorithms are completely equivalent. In general, the computing time of a band selection method is related to three parameters:

TABLE I
Computational Complexity of VGBS and FastVGBS

Algorithm	Number of flops
VGBS	$O\left(\sum_{k=n+1}^{L-1} k^3\right)$
FastVGBS	$O\left(\sum_{k=n+1}^{L-1} k^2\right)$



(a) (b)

Fig. 1. Truth color image (R: Band 329, G: Band 218, B: Band 107) of the Lottery data (a) and the spectral signature of the marked pixel (b).

the number of bands (L), the number of selected bands (n), and the number of pixels (N). A real hyperspectral image, namely the Lottery data, is used to compare the computing time of the two methods with different Ls, ns, and Ns, respectively. The data have a size of 51 × 51 pixels and were collected by the hyperspectral and spatial information system using the Analytical Spectral Devices, Inc. (ASD) FieldSpec spectroradiometer with 2151 bands over the full 350 to 2500 nm spectral range under the laboratory condition on 11 June, 2009. A true-color image of the Lottery data is shown in Fig. 1(a), and the spectral signature of one pixel is shown in Fig. 1(b).

Both the algorithms are implemented using the MATLAB (R2017a) software and based on an Intel Core i7-8850U CPU running at 1.80 GHz with 8.00 GB (RAM). To reduce uncertainty, the average computing time of 20 runs on both algorithms is computed. Similar to Section III, we only collect the computing time of the band selection process of the two algorithms for comparison. For FastVGBS, the computing time only includes the running time of Step 2–7 of Algorithm 1.

First, we compare the computing efficiency of FastVGBS and VGBS with a different number of bands by randomly selecting L bands from the original L_0 (=2151) bands. Here, the number of bands, L, is set from 200 to 2100 with an interval of 100, and the number of selected bands, n, is fixed to $n = 100$.

The computing time of the two methods is shown in Fig. 2(a), and we can find that both the methods require more time as L increases. Still, the growth rate of computing time of VGBS is much faster than that of FastVGBS. It can be seen that although the total size of this data is small, the computing time of VGBS will exceed 10 s when $L \ge 800$.

$$\mathbf{K}_k^{-1} = \begin{bmatrix} \mathbf{K}_{k-1} & \mathbf{u} \\ \mathbf{u}^{\mathrm{T}} & a \end{bmatrix}^{-1} = \begin{bmatrix} \mathbf{K}_{k-1}^{-1} + \mathbf{K}_{k-1}^{-1}\mathbf{u}\left(a - \mathbf{u}^{\mathrm{T}}\mathbf{K}_{k-1}^{-1}\mathbf{u}\right)^{-1}\mathbf{u}^{\mathrm{T}}\mathbf{K}_{k-1}^{-1} & -\mathbf{K}_{k-1}^{-1}\mathbf{u}\left(a - \mathbf{u}^{\mathrm{T}}\mathbf{K}_{k-1}^{-1}\mathbf{u}\right)^{-1} \\ -\left(a - \mathbf{u}^{\mathrm{T}}\mathbf{K}_{k-1}^{-1}\mathbf{u}\right)^{-1}\mathbf{u}^{\mathrm{T}}\mathbf{K}_{k-1}^{-1} & \left(a - \mathbf{u}^{\mathrm{T}}\mathbf{K}_{k-1}^{-1}\mathbf{u}\right)^{-1} \end{bmatrix}. \tag{10}$$

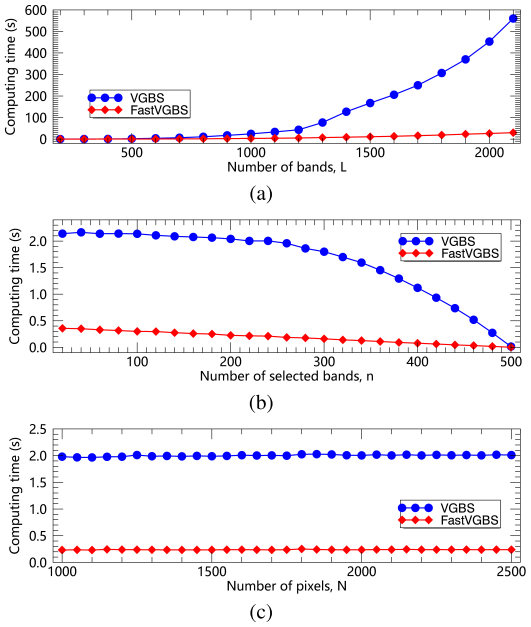


Fig. 2. Computing time of VGBS and FastVGBS for the Lottery data with (a) number of bands, L ranging from 200 to 2100 with an interval of 100 (the number of pixels $= 51 \times 51$ and the number of selected bands $= 100$). (b) Number of selected bands, n ranging from 20 to 500 with an interval of 20 (the number of bands $= 501$ and the number of pixels $= 51 \times 51$). (c) Number of pixels, N ranging from 1000 to 2500 with an interval of 50 (the number of bands $= 501$ and the number of selected bands $= 200$).

Moreover, when $L = 2100$, the computing time of VGBS reaches as high as $T_{\text{VGBS}} = 637.90$ s, whereas the computing time of FastVGBS is $T_{\text{FastVGBS}} = 31.81$ s, which is only 4.99% of T_{VGBS}. This result indicates that VGBS has a very high computational complexity when the number of bands is large. It is because VGBS needs to recalculate the covariance matrix $\mathbf{K}^{-1}$ each time after the band with the maximum gradient being removed. The computational complexity of calculating $\mathbf{K}^{-1}$ is $O(k^3)$, where k is the dimension of $\mathbf{K}$ and is directly related to the number of bands L. As a result, when L is large, the computing time of VGBS becomes unbearable. Since FastVGBS can reduce the complexity of computing $\mathbf{K}^{-1}$ by an order of magnitude, its computing time can be significantly shortened.

Next, we evaluate the computational performance of VGBS and FastVGBS with a different number of selected bands (n). Here, 501 bands (Bands 51 to 551) are used, and n is changed from 20 to 500 with an interval of 20. The computing time curves of VGBS and FastVGBS are shown in Fig. 2(b). It can be found that different from the above experiment, the computing time of both methods decreases as the number of selected bands increases. That is to say, VGBS and FastVGBS require less computing time when selecting more bands. It is because they both use the sequential backward searching strategy. Again, FastVGBS is always faster than VGBS. Moreover, the computational superiority of FastVGBS increases as n decreases, which indicates that FastVGBS has more advantages when selecting fewer bands from a data set.

Finally, the computing time of VGBS and FastVGBS is compared by varying the number of total pixels by randomly selecting N pixels from the original N_0 $(= 51 \times 51)$ pixels ranging from 1000 to 2500 with an interval of 50. Similar to the second experiment, 501 bands (Bands 51 to 551) are used, and the number of selected bands, n, is set to 200.

The result is shown in Fig. 2(c). We can see that the computing time of VGBS remains stable for all Ns. The computing time of FastVGBS shows the same trend but with a much lower level. This result is consistent with the theoretical analysis in Section III. As shown in Table I, the computational complexities of the two algorithms are not related to the number of total pixels, N. As a result, when testing on the real data set, the computing time is not sensitive to N.

V. Conclusion

The VGBS is an effective band selection method for hyperspectral remote sensing data. It uses the maximum volume criterion, and by introducing the relationship between the volume of a subsimplex and the volume gradient of a simplex with respect to hyperspectral images, it successively removes the most redundant band with the largest volume gradient. Still, VGBS needs to recalculate the gradient matrix each time when removing a band, so the computing time of VGBS is still high, especially when applied to data with massive bands. In this letter, 1) by introducing the relation between the norm of each row of the gradient matrix and the diagonal element of the covariance matrix, we have simplified the calculation of the former and 2) by proposing a recursive formula to calculate the inverse of the covariance matrix, we can reduce the computational complexity from $O(L^3)$ to $O(L^2)$ (L is the number of bands). Experiments using real hyperspectral images with a different number of selected bands, different number of bands, and different number of pixels show that FastVGBS is always faster than VGBS and, therefore, can better meet the real-time processing requirement.

References

[1] W. Zhang, X. Li, Y. Dou, and L. Zhao, "A geometry-based band selection approach for hyperspectral image analysis," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 8, pp. 4318–4333, Aug. 2018.

[2] B. Kim and D. A. Landgrebe, "Hierarchical classifier design in high-dimensional numerous class cases," *IEEE Trans. Geosci. Remote Sens.*, vol. 29, no. 4, pp. 518–528, Jul. 1991.

[3] X. Geng, K. Sun, L. Ji, and Y. Zhao, "A fast volume-gradient-based band selection method for hyperspectral image," *IEEE Trans. Geosci. Remote Sens.*, vol. 52, no. 11, pp. 7111–7119, Nov. 2014.

[4] C.-I. Chang and S. Wang, "Constrained band selection for hyperspectral imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 44, no. 6, pp. 1575–1585, Jun. 2006.

[5] C. Sheffield, "Selecting band combinations from multispectral data," *Photogramm. Eng. Remote Sens.*, vol. 58, no. 6, pp. 681–687, 1985.

[6] P. Gao, J. Wang, H. Zhang, and Z. Li, "Boltzmann entropy-based unsupervised band selection for hyperspectral image classification," *IEEE Geosci. Remote Sens. Lett.*, vol. 16, no. 3, pp. 462–466, Mar. 2019.

[7] K. Sun, X. Geng, and L. Ji, "Exemplar component analysis: A fast band selection method for hyperspectral imagery," *IEEE Geosci. Remote Sens. Lett.*, vol. 12, no. 5, pp. 998–1002, May 2015.

[8] S. Jia, G. Tang, J. Zhu, and Q. Li, "A novel ranking-based clustering approach for hyperspectral band selection," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 1, pp. 88–102, Jan. 2015.

[9] X. Cao, B. Wu, D. Tao, and L. Jiao, "Automatic band selection using spatial-structure information and classifier-based clustering," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 9, no. 9, pp. 4352–4360, Sep. 2017.

[10] B. Dennis, *Matrix Mathematics*. Princeton, NJ, USA: Princeton Univ. Press, 2005.

[11] R. Archibald and G. Fann, "Feature selection and classification of hyperspectral images with support vector machines," *IEEE Geosci. Remote Sens. Lett.*, vol. 4, no. 4, pp. 674–677, Oct. 2007.