GRAPH LEARNING BASED AUTOENCODER FOR HYPERSPECTRAL BAND SELECTION

Yongshan Zhang[1,2]*, Xinxin Wang*[2]*, Zhenyu Wang*[1]*, Xinwei Jiang*[1]*, Yicong Zhou*[2]

[1] School of Computer Science, China University of Geosciences, Wuhan, 430074, China
[2] Department of Computer and Information Science, University of Macau, Macau, 999078, China

ABSTRACT

Hyperspectral band selection aims to identify an optimal subset of bands from hyperspectral images (HSIs). Most existing methods explore the relationships between pair-wise pixels in a fixed graph. However, the quality of the initial fixed graph may be influenced by noises and user-defined parameters that may not be optimal for HSI analysis. In this paper, we propose a graph learning based autoencoder (GLAE) to achieve unsupervised hyperspectral band selection. Using the relationships of pair-wise pixels within HSIs, GLAE constructs the initial graph to characterize the geometric structures of HSIs and then adjusts the graph to adapt the band selection process. To solve the proposed model, we intoduce an alternative optimization algorithm. Experiments and comparisons on three HSI datasets demonstrate that the proposed GLAE achieves better results over the state-of-the-art methods.

Index Terms— Hyperspectral image, band selection, autoencoder, graph learning

1. INTRODUCTION

With advances in spectral imaging technology, hyperspectral images (HSIs) become prevalent in the field of remote sensing [1, 2]. An HSI is generally represented by a cube that provides plentiful spectral and spatial information to realize earth observation tasks [3, 4]. The pixels in an HSI are usually described by hundreds of spectral bands that may contain noisy, irrelevant and redundant information. With the increased number of spectral bands, the inconvenience and inefficiency emerge in storage and processing. Besides, high-dimensional spectral bands may result in the Hughes phenomenon and degraded performance in HSI applications.

To avoid these problems, it is desirable to retain helpful bands and remove harmful ones to learn HSIs without losing effectiveness. This technique is also known as band selection (i.e., feature selection), that has attracted much attention and extensive efforts in remote sensing community [5, 6]. According to the availability of label information, band selection can be realized in supervised, semi-supervised and unsupervised manner [7]. Supervised and semi-supervised band selection utilize complete labels or limited labels to discriminate informative bands. Different from the two categories, unsupervised band selection selects a subset of representative bands without using any explicit labels [8, 9, 10]. Considering the labor-intensive and time-consuming annotation, unsupervised style is more flexible and preferred in most cases. There are numerous effective unsupervised band selection methods. Most of them are generally designed as linear models. For example, self-representation assumes that the original data can be represented as a linear combination of itself [11]. In reality, it is difficult to extract the nonlinear characteristics from HSIs by linear learning models.

In recent years, autoencoder has been widely studied for unsupervised feature selection. Due to the nonlinear activation function and encoder-decoder architecture, it can be regarded as a nonlinear self-representation model [12, 13]. By integrating a feature selection regularization term, autoencoders can be used to exploit nonlinear relationships among features and conduct feature selection, e.g., feature selection guided autoencoder [14], autoencoder feature selection [15] and graph regularized autoencoder feature selection [16]. These methods have achieved a great success. The graph introduced in these methods is fixed and unchangeable during learning that may not be optimal for HSI analysis. Thus, they can not be well adapted to hyperspectral band selection.

Based on the above observations, in this paper, we propose a graph learning based autoeocnder (GLAE) for unsupervised hyperspectral band selection. In GLAE, graph learning is considered into the autoencoder-based feature selection model. Specifically, using the relationships between pair-wise pixels within an HSI, the initial graph is constructed to characterize the geometric structures of an HSI and then adjusted to promote unsupervised band selection. To solve the proposed model, we advance an alternative optimization algorithm as solution. Experiments on three HSI datasets validate the advantage and potential of the proposed GLAE.

The structure of this paper is organized as follows. Section 2 firstly presents the formulation of the proposed GLAE method and then provides an alternative optimization algorithm. Experiments and comparisons on three HSI datasets are demonstrated in Section 3. Finally, we conclude the paper in Section 4.

This work was supported in part by the National Natural Science Foundation of China under Grant 62106241 and in part by the Natural Science Foundation of Hubei Province of China under Grant 2020CFB328.

2. PROPOSED METHOD

2.1. Formulation

Autoencoder is an unsupervised neural network framework with the same neurons in both input and output layers [17]. It learns an auxiliary representation from the inputs and then reconstruct another representation that approximates to the original inputs as possible [18]. By imposing a row-sparsity regularizer, autoencoder can be adapted for feature selection. Given a hyperspectral data matrix $\mathbf{X} \in \mathbb{R}^{N\times d}$ (reshaped from an HSI cube) with N pixels and d spectral bands, the autoencoder feature/band selector is formulated as

$$\min_{\mathcal{H}} ||\mathbf{X}-g(f(\mathbf{X}))||_F^2+\alpha||\mathbf{W}^{(1)}||_{2,1}+\beta\sum_{i=1}^{2}||\mathbf{W}^{(i)}||_F^2, \quad (1)$$

where $f(\mathbf{X}) = \sigma_1(\mathbf{W}^{(1)}\mathbf{X} + \mathbf{B}^{(1)})$ denotes the encoder, and $\hat{\mathbf{X}} = g(f(\mathbf{X})) = \sigma_2(\mathbf{W}^{(2)}f(\mathbf{X}) + \mathbf{B}^{(2)})$ represents the decoder. $\sigma_1(\cdot)$ and $\sigma_2(\cdot)$ are the predefined activation functions. $\mathcal{H} = \{\mathbf{W}^{(1)}, \mathbf{W}^{(2)}, \mathbf{B}^{(1)}, \mathbf{B}^{(2)}\}$ are the network weights and layer biases for the autoencoder. Parameters α and β are used to balance the $\ell_{2,1}$-norm constraint and weight decay regularization. $||\mathbf{W}^{(1)}||_{2,1}$ produces a row-sparse matrix to measure the importance of each band and $\sum_{i=1}^{2}||\mathbf{W}^{(i)}||_F^2$ takes a role in avoiding overfitting and promoting convergence.

To describe the relationships between pair-wise pixels, the initial graph $\mathbf{A} \in \mathbb{R}^{N\times N}$ is constructed by a k-nearest neighbor graph. However, the quality of the initial graph is greatly affected by noises that may result in unsatisfactory performance [19]. To alleviate this problem, we adopt a graph learning mechanism to learn an optimal graph $\mathbf{S} \in \mathbb{R}^{N\times N}$ based on $\mathbf{A}$ with some desirable properties [20]. Thus, the formulation of graph learning is given by

$$\min_{\mathbf{S}} \|\mathbf{S}-\mathbf{A}\|_F^2, \ s.t.\ \mathbf{S1} = \mathbf{1}, \mathbf{S} \geq \mathbf{0}, \mathrm{rank}(\mathbf{L_S}) = n-c, \quad (2)$$

where $\mathbf{S} \geq \mathbf{0}$ denotes the nonnegative constraint and $\mathbf{S1} = \mathbf{1}$ represents the row-sum-to-one constraint. $\mathrm{rank}(\mathbf{L_S}) = n - c$ indicates the rank constraint for the graph Laplacian $\mathbf{L_S}$ if $\mathbf{S}$ is expected to be exactly c block diagonals. It means that n samples should be partitioned into c clusters. During the learning process, the optimal graph is learnt from the initial graph that can better serve for band selection. Compared to the initial graph, the learnt optimal graph is more robust to outliers and noises.

Based on Ky Fan's Theorem [21], the rank constraint on $\mathbf{L_S}$ can be formulated as the optimization on the pseudo clustering indicator matrix $\mathbf{F}$. For autoencoder, the original data is reconstructed that can learn a discriminative representation $f(\mathbf{X})$ from a new feature space. During this learning process, the pseudo clustering indicator matrix $\mathbf{F}$ can not be obtained. Therefore, we use $f(\mathbf{X})$ instead of $\mathbf{F}$ to characterize the local data structure. Then, Eq. (2) can be rewritten as

$$\begin{aligned}\min_{\mathbf{S},\mathbf{F}} \ & \|\mathbf{S}-\mathbf{A}\|_F^2 + \lambda\,\mathrm{Tr}(f(\mathbf{X})^T\mathbf{L_S}f(\mathbf{X})), \\ & s.t.\ \mathbf{S1} = \mathbf{1}, \mathbf{S} \geq \mathbf{0}.\end{aligned} \quad (3)$$

where λ is the regularization parameter. By simultaneously performing graph learning and autoencoder feature selection, the objective function of GLAE is formulated as

$$\begin{aligned}\min_{\mathcal{H},\mathbf{S}} \ & \|\mathbf{X}-\hat{\mathbf{X}}\|_F^2 + \alpha\|\mathbf{W}^{(1)}\|_{2,1} + \beta\sum_{i=1}^{2}\|\mathbf{W}^{(i)}\|_F^2 \\ & + \gamma\|\mathbf{S}-\mathbf{A}\|_F^2 + \lambda\,\mathrm{Tr}(f(\mathbf{X})^T\mathbf{L_S}f(\mathbf{X})) \\ & s.t.\ \mathbf{S1} = \mathbf{1}, \mathbf{S} \geq \mathbf{0}\end{aligned} \quad (4)$$

where γ is the parameter for trade-off. Once $\mathbf{W}^{(1)}$ is learned, band selection can be realized by ranking bands according to $||\boldsymbol{w}_i^{(1)}||_2$ $(i = 1, ..., d)$ in a descending order.

2.2. Optimization

It is hard to solve Eq. (4) for the nonlinearity in autoencoder. Here, we propose an optimization algorithm to iteratively update one variable by fixing all others. The pseudocode of the proposed GLAE is exhibited in Algorithm 1.

Update S. With other fixed variables, the optimization w.r.t $\mathbf{S}$ becomes

$$\min_{\mathbf{S}1=\mathbf{1},\mathbf{S}\geq\mathbf{0}} \gamma\|\mathbf{S}-\mathbf{A}\|_F^2 + \lambda\,\mathrm{Tr}(f(\mathbf{X})^T\mathbf{L_S}f(\mathbf{X})), \quad (5)$$

which can be represented in scalar form as

$$\min_{\substack{\sum_j s_{ij}=1,\\ s_{ij}\geq 0}} \ \gamma\sum_{i,j=1}^{n}(s_{ij}-a_{ij})^2 + \lambda\sum_{i,j=1}^{n}\|f(\mathbf{x}_i)-f(\mathbf{x}_j)\|_2^2 s_{ij}. \quad (6)$$

Denote $\mathbf{d}_i$ is a vector, where $d_{ij} = \|f(\mathbf{x}_i) - f(\mathbf{x}_j)\|_2^2$ is the jth element. Similarly, Eq. (6) is rewritten in vector form

$$\min_{\mathbf{s}_i\mathbf{1}=1,\mathbf{s}_i\geq\mathbf{0}} \left\|\mathbf{s}_i - \left(\mathbf{a}_i - \frac{\lambda}{2\gamma}\mathbf{d}_i\right)\right\|_F^2. \quad (7)$$

This optimization problem can be finally resolved with a closed form solution by an iterative algorithm [22].

To update $\mathbf{W}^{(1)}, \mathbf{W}^{(2)}, \mathbf{B}^{(1)}, \mathbf{B}^{(2)}$, the error terms for the output and hidden layers are necessary for calculating the derivatives the four variables. In detail, the error terms of the output and hidden layers are determined by

$$\delta^{(o)} = 2(\mathbf{X} - g(f(\mathbf{X}))) \odot g'(f(\mathbf{X})); \quad (8)$$

$$\delta^{(h)} = (\delta^{(o)}{\mathbf{W}^{(2)}}^T) \odot f'(\mathbf{X}); \quad (9)$$

where $\odot$ indicates the element-wise product. With the two error terms, the derivatives w.r.t the four variables can be determined by a chain rule and back-propagation strategy.

Algorithm 1 GLAE

Input: Hyperspectral data $\mathbf{X} \in \mathbb{R}^{N\times d}$, initial graph matrix $\mathbf{A} \in \mathbb{R}^{N\times N}$, parameters α, β, γ and λ;
Output: Top ranked bands;

```
Initialize W^(1), W^(2), B^(1), B^(2) and S;
repeat
    Calculate δ^(o) and δ^(h) by Eqs. (8) and (9);
    Update S by solving Eq. (7);
    Calculate the derivateives of W^(1), W^(2), B^(1) and
    B^(2) according to Eqs. (11), (13), (15) and (17);
    Update W^(1), W^(2), B^(1) and B^(2) via Eq. (18);
    Update D as d_ii = 1/(2||w_i^(1)||_2);
until Convergence
Select the top ranked bands with large values of ||w_i^(1)||_2.
```

Derivative w.r.t $\mathbf{W}^{(2)}$. With other fixed variables, the optimization w.r.t $\mathbf{W}^{(2)}$ becomes

$$\mathcal{J}(\mathbf{W}^{(2)}) = ||\mathbf{X} - \hat{\mathbf{X}}||_F^2 + \beta||\mathbf{W}^{(2)}||_F^2. \tag{10}$$

The derivative w.r.t $\mathbf{W}^{(2)}$ is given by

$$\partial\mathcal{J}/\partial\mathbf{W}^{(2)} = \mathbf{Z}^{(o)^T}\delta^{(o)} + \beta\mathbf{W}^{(2)}. \tag{11}$$

Derivative w.r.t $\mathbf{B}^{(2)}$. With other fixed variables, the optimization w.r.t $\mathbf{B}^{(2)}$ becomes

$$\mathcal{J}(\mathbf{B}^{(2)}) = ||\mathbf{X} - \hat{\mathbf{X}}||_F^2. \tag{12}$$

The derivative w.r.t $\mathbf{B}^{(2)}$ is given by

$$\partial\mathcal{J}/\partial\mathbf{B}^{(2)} = \delta^{(o)}. \tag{13}$$

Derivative w.r.t $\mathbf{W}^{(1)}$. With other fixed variables, the optimization w.r.t $\mathbf{W}^{(1)}$ becomes

$$\begin{aligned}\mathcal{J}(\mathbf{W}^{(1)}) = &||\mathbf{X} - \hat{\mathbf{X}}||_F^2 + \alpha||\mathbf{W}^{(1)}||_{2,1} \\ &+ \beta||\mathbf{W}^{(1)}||_F^2 + \lambda\,\mathrm{Tr}(f(\mathbf{X})^T\mathbf{L}_\mathbf{S}f(\mathbf{X})).\end{aligned} \tag{14}$$

The derivative w.r.t $\mathbf{W}^{(1)}$ is given by

$$\begin{aligned}\partial\mathcal{J}/\partial\mathbf{W}^{(1)} =&\mathbf{X}^T\delta^{(h)} + \alpha\mathbf{D}\mathbf{W}^{(1)} + \beta\mathbf{W}^{(1)} \\ &+ 2\lambda\mathbf{X}^T(\mathbf{L}_\mathbf{S}f(x)\odot f'(\mathbf{X})).\end{aligned} \tag{15}$$

In Eq. (15), $\mathbf{D}$ is a diagonal matrix, where each diagonal element is updated by $d_{ii} = \frac{1}{2||\boldsymbol{w}_i^{(1)}||_2}$ in each iteration.

Derivative w.r.t $\mathbf{B}^{(1)}$. With other fixed variables, the optimization w.r.t $\mathbf{B}^{(1)}$ becomes

$$\mathcal{J}(\mathbf{B}^{(1)}) = ||\mathbf{X} - \hat{\mathbf{X}}||_F^2 + \lambda\,\mathrm{Tr}\left(f(\mathbf{X})^T\mathbf{L}_\mathbf{S}f(\mathbf{X})\right). \tag{16}$$

The derivative w.r.t $\mathbf{W}^{(1)}$ is given by

$$\partial\mathcal{J}/\partial\mathbf{B}^{(1)} = \delta^{(h)} + 2\lambda\mathbf{L}_\mathbf{s}f(x)\odot f'(\mathbf{X}). \tag{17}$$

(a) Indian Pines (b) PaviaU (c) Salinas

Fig. 1. The false color composition of three HSI datasets.

Update $\mathbf{W}^{(1)}$, $\mathbf{W}^{(2)}$, $\mathbf{B}^{(1)}$ and $\mathbf{B}^{(2)}$. Given the above derivatives, the update rules for $\mathbf{W}^{(1)}$, $\mathbf{W}^{(2)}$, $\mathbf{B}^{(1)}$ and $\mathbf{B}^{(2)}$ are provided by the gradient descent as

$$\begin{aligned}\mathbf{W}^{(1)} &= \mathbf{W}^{(1)} - \eta\frac{\partial\mathcal{J}}{\partial\mathbf{W}^{(1)}}, \quad \mathbf{B}^{(1)} = \mathbf{B}^{(1)} - \eta\frac{\partial\mathcal{J}}{\partial\mathbf{B}^{(1)}}, \\ \mathbf{W}^{(2)} &= \mathbf{W}^{(2)} - \eta\frac{\partial\mathcal{J}}{\partial\mathbf{W}^{(2)}}, \quad \mathbf{B}^{(2)} = \mathbf{B}^{(2)} - \eta\frac{\partial\mathcal{J}}{\partial\mathbf{B}^{(2)}},\end{aligned} \tag{18}$$

where η is the learning rate. To sum up, all variables can be iteratively updated in the optimization algorithm.

3. EXPERIMENTS

3.1. Experimental Setup

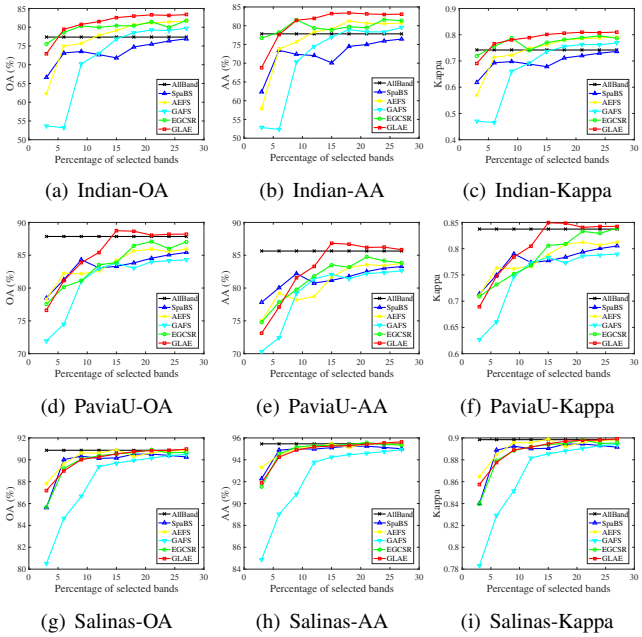
Datasets. To verify the performance of the proposed GLAE, experimental studies are conducted on three HSI datasets. The Indian Pine dataset contains 145×145 pixels and 16 land covers. Each pixel is described by 200 useful spectral bands. The University of Pavia (PaviaU) dataset with 610×340 pixels is represented by 103 valid spectral bands and associated with 9 land covers. The Salinas Scene dataset has a spatial dimension of 512×217 pixels. There are 204 informative spectral bands to describe 16 land covers. The false color composition for the three datasets are shown in Fig. 1.

Comparison methods. The state-of-the-art methods are used to make holistic comparisons with the proposed GLAE. SpaBS is a sparse representation method to select discriminative bands [11]. AEFS combines autoencoder and group lasso task in a unified framework [15]. GAFS considers spectral graph analysis into autoencoder feature selector [16]. EGCSR selects representative bands with graph convolution [23]. AllBand is a baseline method using all spectral bands.

Parameter Settings. In GLAE, parameters α, β, γ and λ are tuned from $\{10^{-3}, 10^{-2}, ..., 10^{3}\}$ by a grid-search strategy. Since the sizes of spectral bands are not very large, the numbers of hidden neurons are tuned from $\{5, 15, 25, ..., 95\}$ without exceeding the original dimension. For band selection, the desired number of selected bands is unknown and difficult to determine in practice for different datasets. Thus, we vary the percentages of selected bands from $\{3\%, 6\%, ..., 27\%\}$. For a fair comparison, k-nearest neighbors (KNN) [24] is

Table 1. Comparisons of different methods on three HSI datasets.

Dataset	Measure	AllBand	SpaBS	AEFS	GAFS	EGCSR	GLAE
Indian Pines	OA(%)	77.37	76.91	81.66	79.73	81.81	**83.37**
	AA(%)	77.83	76.43	81.26	79.52	81.64	**83.41**
	Kappa	0.7418	0.7363	0.7908	0.7686	0.7957	**0.8103**
PaviaU	OA(%)	87.86	85.43	85.97	84.31	87.09	**88.74**
	AA(%)	85.64	83.28	83.74	82.63	84.75	**86.82**
	Kappa	0.8373	0.8052	0.8126	0.7898	0.8391	**0.8494**
Salinas	OA(%)	90.87	90.52	90.91	90.49	90.93	**90.97**
	AA(%)	95.46	95.25	95.48	94.93	95.58	**95.64**
	Kappa	0.8984	0.8944	0.8989	0.8941	0.8983	**0.8992**



(a) Indian-OA (b) Indian-AA (c) Indian-Kappa

(d) PaviaU-OA (e) PaviaU-AA (f) PaviaU-Kappa

(g) Salinas-OA (h) Salinas-AA (i) Salinas-Kappa

Fig. 2. Comparisons of different methods w.r.t different numbers of selected bands on three HSI datasets.

adopted as the classifier to verify the quality of selected bands. We randomly choose 10% samples of each class for training and the rest 90% samples are used for testing. Three popular measures, including overall accuracy (OA), average accuracy (AA) and kappa coefficient, are adopted for evaluation. The best results with optimal parameters are presented in the following section.

3.2. Experimental Results

The best classification results of all competing methods on the three HSI datasets are presented in Table 1. As evident from Table 1, the proposed GLAE consistently outperforms all compared methods. Compared to AllBand, the improvement of GLAE on the Indian Pines dataset is obvious and high at 7.75%, 7.17% and 9.23% for OA, AA and Kappa, respectively. Learning with the structural information, EGCSR shows the second best results on Indian Pine dataset, but it is also inferior to GLAE. The above observations demonstrate that GLAE is effective in selecting representative spectral bands. On the PaviaU dataset, GLAE is superior to AllBand, and SpaBS, AEFS and GAFS are inferior to AllBand. This indicates that SpaBS, AEFS and GAFS may remain the noisy bands during the band selection process. GAFS is degraded version of GLAE without graph learning mechanism. The comparison between GLAE and GAFS can be regarded as an ablation study. The experimental results also verify the effectiveness of graph learning mechanism in GLAE.

The results of all competing methods w.r.t different numbers of selected bands are shown in Fig. 2. With an increasing number of selected bands, the performance first tends to increase and then keeps stable or even degrades. This indicates that these methods with an improper number of selected bands may ignore valuable bands or remain noisy bands. With different numbers of selected bands, GLAE outperforms other methods in majority. Compared to others, GLAE achieves superior results on the Indian Pines and PaviaU datasets. This can be also explained that GLAE is effective in selecting representative bands with the help of graph learning.

4. CONCLUSION

In this paper, we proposed a graph learning based autoencoder (GLAE) for unsupervised hyperspectral band selection. To select representative bands from HSIs, GLAE constructs an initial graph to model the relationships between pair-wise pixels and conducts band selection accompanied by graph learning. The graph learning process improves the quality of the graph that can achieve better performance. An effective optimization algorithm is devised to solve the proposed GLAE. Experiments on three HSI datasets demonstrated that GLAE is effective for hyperspectral band selection and outperforms the state-of-the-art methods. Since the proposed GLAE is a single-hidden layer model, it can be easily used as a building block to form a hierarchy. In the future, we will investigate deep architectures for hyperspectral band selection.

5. REFERENCES

[1] M. Fauvel, Y. Tarabalka, J. A. Benediktsson, J. Chanussot, and J. C. Tilton, "Advances in spectral-spatial classification of hyperspectral images," *Proceedings of the IEEE*, vol. 101, no. 3, pp. 652–675, 2012.

[2] X. Zhang, X. Jiang, J. Jiang, Y. Zhang, X. Liu, and Z. Cai, "Spectral-spatial and superpixelwise pca for unsupervised feature extraction of hyperspectral imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, pp. 1–10, 2022.

[3] D. Lunga, S. Prasad, M. M. Crawford, and O. Ersoy, "Manifold-learning-based feature extraction for classification of hyperspectral data: A review of advances in manifold learning," *IEEE Signal Proc. Mag.*, vol. 31, no. 1, pp. 55–66, 2014.

[4] G. Fan, Y. Ma, J. Huang, X. Mei, and J. Ma, "Robust graph autoencoder for hyperspectral anomaly detection," in *IEEE Int. Conf. Acoust. Speech Signal Process.*, 2021, pp. 1830–1834.

[5] W. Sun, L. Zhang, B. Du, W. Li, and Y. M. Lai, "Band selection using improved sparse subspace clustering for hyperspectral imagery classification," *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.*, vol. 8, no. 6, pp. 2784–2797, 2015.

[6] L. Shen, Z. Zhu, S. Jia, J. Zhu, and Y. Sun, "Discriminative gabor feature selection for hyperspectral image classification," *IEEE Geosci. Remote Sens. Lett.*, vol. 10, no. 1, pp. 29–33, 2012.

[7] F. He, F. Nie, R. Wang, W. Jia, F. Zhang, and X. Li, "Semisupervised band selection with graph optimization for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 12, pp. 10298–10311, 2021.

[8] Y. Yuan, X. Zheng, and X. Lu, "Discovering diverse subset for unsupervised hyperspectral band selection," *IEEE Trans. Image Process.*, vol. 26, no. 1, pp. 51–64, 2016.

[9] G. Zhu, Y. Huang, J. Lei, Z. Bi, and F. Xu, "Unsupervised hyperspectral band selection by dominant set extraction," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 1, pp. 227–239, 2015.

[10] Y. Zhang, X. Wang, X. Jiang, and Y. Zhou, "Marginalized graph self-representation for unsupervised hyperspectral band selection," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, pp. 1–12, 2022.

[11] S. Li and H. Qi, "Sparse representation based band selection for hyperspectral images," in *Proc. IEEE Int. Conf. Image Process.*, 2011, pp. 2693–2696.

[12] S. C. AP, S. Lauly, H. Larochelle, M. Khapra, B. Ravindran, V. C. Raykar, and A. Saha, "An autoencoder approach to learning bilingual word representations," in *Proc. Advances Neural Inf. Process. Syst.*, 2014, pp. 1853–1861.

[13] S. Ozkan, B. Kaya, and G. B. Akar, "Endnet: Sparse autoencoder network for endmember extraction and hyperspectral unmixing," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 1, pp. 482–496, 2018.

[14] S. Wang, Z. Ding, and Y. Fu, "Feature selection guided auto-encoder," in *Proc. AAAI Conf. Artif. Intell.*, 2017, pp. 2725–2731.

[15] K. Han, Y. Wang, C. Zhang, C. Li, and C. Xu, "Autoencoder inspired unsupervised feature selection," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process.*, 2018, pp. 2941–2945.

[16] S. Feng and M. F. Duarte, "Graph regularized autoencoder-based unsupervised feature selection," in *Proc. Asilomar Conf. Signals. Syst. Comput.*, 2018, pp. 55–59.

[17] G. E. Hinton and R. R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *Science*, vol. 313, no. 5786, pp. 504–507, 2006.

[18] A. Majumdar, "Blind denoising autoencoder," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 30, no. 1, pp. 312–317, 2018.

[19] F. Nie, W. Zhu, and X. Li, "Unsupervised feature selection with structured graph optimization," in *Proc. AAAI Conf. Artif. Intell.*, 2016, pp. 1302–1308.

[20] F. Nie, X. Wang, M. Jordan, and H. Huang, "The constrained laplacian rank algorithm for graph-based clustering," in *Proc. AAAI Conf. Artif. Intell.*, 2016.

[21] K. Fan, "On a theorem of weyl concerning eigenvalues of linear transformations," *Proceedings of the National Academy of Sciences*, vol. 35, no. 11, pp. 652, 1949.

[22] J. Huang, F. Nie, and H. Huang, "A new simplex sparse learning model to measure data similarity for clustering," in *Proc. Int. Joint Conf. Artif. Intell.*, 2015, pp. 3569–3575,.

[23] Y. Cai, Z. Zhang, X. Liu, and Z. Cai, "Efficient graph convolutional self-representation for band selection of hyperspectral image," *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.*, vol. 13, pp. 4869–4880, 2020.

[24] Hanan Samet, "K-nearest neighbor finding using maxnearestdist," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 30, no. 2, pp. 243–252, 2007.