# Graph Attention Convolutional Autoencoder-Based Unsupervised Nonlinear Unmixing for Hyperspectral Images

Danni Jin 🆔 and Bin Yang 🆔, *Member, IEEE*

*Abstract*—Hyperspectral unmixing has received increasing attention as a technique for estimating endmember spectra and fractional abundances of land covers. Encoding high-dimensional hyperspectral data into a low-dimensional latent space to generate reasonable abundances, autoencoder (AE) has shown its great potential and attractive advantages in spectral unmixing. AEs decode abundances back to spectra, which can effectively reflect the general spectral mixing process. However, most existing AE-based unmixing methods often do not fully exploit the spatial information of hyperspectral images, hindering the improvement of unmixing accuracy. This article proposes a graph attention convolutional autoencoder architecture for hyperspectral unmixing. By incorporating graph attention convolution into AE, the proposed method performs better in leveraging both long-range and short-range spatial information of hyperspectral images. Accurate abundances with global and local spatial consistency can be efficiently learned by the network. Moreover, the decoder is further improved based on the postpolynomial nonlinear mixing model to make the network have stronger physical interpretability to deal with the issue of nonlinear blind unmixing. Experimental results indicate that the proposed method has good unmixing performance. It can reduce the root mean squared error of estimated abundances for synthetic data by over 10% compared to other methods. In experiments with real hyperspectral data, the difference between its unmixing results' accuracy and the best is less than 5%.

*Index Terms*—Autoencoder, bilinear mixture model, graph attention convolutional network, hyperspectral remote sensing imagery, nonlinear spectral unmixing.

## I. INTRODUCTION

**H**YPERSPECTRAL remote sensors capture the reflectance of the earth's surface across a wide range of wavelengths, typically in the visible, near-infrared, and short-wave infrared regions of the electromagnetic spectrum. Collected from platforms such as aircraft and satellites, hyperspectral images play an essential role in a variety of applications, such as land cover classification [1], [2], [3] target detection [4], [5], feature extraction [6], etc. However, problems such as the low resolution of imaging instruments have always led to the prevalence of mixed pixels, i.e., multiple substances exist in a single pixel [7]. Therefore, hyperspectral unmixing techniques are often required to solve this problem to obtain fine land cover information from hyperspectral data [8].

In the past few decades, hyperspectral unmixing has become a research hot spot in the field of hyperspectral image processing. Typical unmixing methods can be divided into two categories: 1) supervised methods; and 2) unsupervised methods. Supervised hyperspectral unmixing methods require a set of known endmembers to be provided to guide the regression of abundances in unmixing. Unsupervised methods automatically update both endmembers and abundances during the unmixing process. Typical blind source separation approaches such as independent component analysis (ICA) and nonnegative matrix factorization (NMF) are popularly adopted to develop unsupervised unmixing algorithms in literature [9], [10], [11]. With the development of artificial intelligence and machine learning technologies, supervised or unsupervised methods based on deep learning (DL) have also been widely used for hyperspectral unmixing [12], [13], [14], [15]. The DL-based unmixing methods' main advantage is that they perform better in learning valuable hidden features of abundances from hyperspectral data directly without explicit dependence on complex mathematical models [16].

Deep autoencoder (AE) networks are especially suitable for unmixing and have shown excellent performance. AEs cannot only extract robust features from high-dimensional hyperspectral images with different data distributions and capture features at different scales, but also can be easily integrated with other advanced deep learning modules. For example, Su et al. [17] proposed an unmixing method by using stacked nonnegative autoencoders to remove outliers in the original data whose reconstructed data were further fed into variational autoencoders (VAE). Similarly, the idea of combining multiple autoencoders to improve unmixing is also adopted in some recently published works. Gao et al. [18] used two autoencoders to constitute an end-to-end cascaded unmixing network model where a self-perception loss was further used to enhance the unmixing performance. Two concatenated autoencoders were introduced to better find the global optimum in [19]. Hua et al. [20] added constraints on the basis of AE to enhance the smoothness of abundances. Two branches of dual-branch autoencoder network (DBA) proposed in [21] used fully connected layers and 2-D

The authors are with the School of Computer Science and Technology, Donghua University, Shanghai 201620, China (e-mail: dnj19980125 @gmail.com; yangb19@dhu.edu.cn).

convolutional layers to extract spectral and spatial information, respectively, which can improve the efficiency and accuracy of feature extraction.

It is noted that most of the AE-based unmixing methods are based on the assumption of the linear mixture model (LMM). The LMM often describes the phenomenon of material mixing at a macroscale, which assumes that the sensor directly receives the incident light after only interacting with a single surface object. However, in many real scenarios, the spectrum of a pixel is significantly affected by the light's multiple scattering and reflection among adjacent objects [22]. Therefore, it is necessary to explain the nonlinear mixture components of pixels with a reasonable mixture model [23]. Representative traditional physical models such as the Hapke model [24] and the PROSPECT model [25], have complex mathematical forms and depend on the acquisition of physical parameters. To decompose the nonlinear mixtures while maintaining the physical meaning, simplification has been carried out, leading to bilinear mixture models [26] such as the polynomial posterior nonlinear mixing (PPNM) model [27], higher-order linear mixture models, and kernel models that describe different degrees of multiple scattering effects [28].

In order to use AE to improve nonlinear unmixing, researchers [29] tried to build AE-based unmixing networks by considering typical numerical nonlinear mixture models such as the PPNM model in the decoder part. Hapke model and fully convolutional neural network (CNN) were recently combined to construct the AE-based nonlinear unmixing model [30]. Zhao et al. considered every pixel of a hyperspectral image as an additive mixture of linear and a nonlinear-fluctuation [31], and they recently introduced 3-D convolution to jointly utilize spatial and spectral information in nonlinear unmixing [32]. Due to the complexity of nonlinear unmixing, similar studies resorting to 2-D or 3-D convolution modules have been done to use spatial information of hyperspectral images to improve unmixing accuracy [33], [34], [35]. Moreover, a feasible and common approach is to build a spatial information-driven constraint term of loss function [36].

The traditional convolution modules can learn spatial information in local rectangular blocks. However, the dependence on the regular convolution kernel and its sizes and the network layers' depth hinders the traditional convolution efficiently learning the complex spatial distribution of land covers. Therefore, to improve the AE-based nonlinear unmixing, it is meaningful and challenging to reasonably construct the network structure to not only better utilize the local and global spatial information of hyperspectral data, but also have physical interpretability like nonlinear mixture models. This is the leading research intention of this work.

In this article, we propose a graph attention convolutional autoencoder (GACAE) for unsupervised nonlinear unmixing. Besides the local spatial information of hyperspectral data learned by traditional CNN modules, the graph attention convolution mechanism is further adopted to learn hyperspectral data's global spatial information. First, a group of spatial blocks is automatically generated by superpixel segmentation of hyperspectral data [37], [38]. Each superpixel block is regarded as a graph node and they can be connected, which allows for the acquisition of

global spatial information through graph convolution networks (GCNs) [39], and computational complexity is much lower than the methods that treat each pixel as a separate graph node [40], [41]. Second, to improve GCNs, a graph attention network (GAT) [42] is introduced, which adaptively adjusts the weights between graph nodes so that the relationship between superpixel blocks can be better reflected. In this sense, it is promising that the spatial context of hyperspectral images in line with the ground truth can be comprehensively learned in the encoding stage. Finally, the PPNM is employed to improve the decoder part of the autoencoder, which enables the network to better explain the nonlinear mixing effects and produce more accurate and physically meaningful unmixing results.

The main contributions of this article are as follows.
1) Superpixel-based GCNs can effectively encode the hyperspectral data's global spatial information with low computational complexity. Moreover, GAT enables more reasonable weight allocation within the graph, leading to further improvement in unmixing accuracy.
2) The global information obtained by GAT is combined with the local information obtained by CNN to fully utilize the spatial information of hyperspectral images.
3) The PPNM is applied in the decoder to make the network physically meaningful for nonlinear unmixing. Compared to the current nonlinear unmixing approaches, the proposed method has shown competitive unmixing performance.

The rest of this article is organized as follows. Section II introduces the principle of AE and the PPNM model. In Section III, the proposed method is presented in detail. The performance of the proposed method on synthetic and real hyperspectral datasets is evaluated in Section IV. Finally, Section V concludes the article and provides a perspective for future work.

## II. RELATED WORKS

### A. AE-Based Unmixing

According to the LMM, an observed pixel $\boldsymbol{x} \in \mathbb{R}^B$ consisting of $B$ spectral bands can be expressed as follows:

$$\boldsymbol{x} = \boldsymbol{M}\boldsymbol{s} + \boldsymbol{\varepsilon} \tag{1}$$

where $\boldsymbol{M} = (\boldsymbol{m}_1, \boldsymbol{m}_2, \ldots, \boldsymbol{m}_R) \in \mathbb{R}^{B \times R}$ denotes the endmember matrix with each column $\boldsymbol{m}_i$ being an endmember spectral signature vector. $\boldsymbol{s} = (\mathrm{s}_1, \mathrm{s}_2, \ldots, \mathrm{s}_R)^{\mathrm{T}}$ and $\boldsymbol{\varepsilon} \in \mathbb{R}^B$ represents the abundance vector and the noise vector associated with the pixel $\boldsymbol{x}$, respectively. In the nonlinear mixing case, $\boldsymbol{x}$ can be denoted as follows:

$$\boldsymbol{x} = \varphi(\boldsymbol{M}, \boldsymbol{s}) + \boldsymbol{\varepsilon} \tag{2}$$

where $\varphi(\cdot)$ represents the nonlinear influence.

An autoencoder network consists of an encoder and a decoder [43]. In (3), $\boldsymbol{S} = (\boldsymbol{s}_1, \boldsymbol{s}_2, \ldots, \boldsymbol{s}_N)$ refers to the high-order hidden features generated by the encoder, which can represent the physically meaningful abundance matrix of the hyperspectral

image, and $N$ represents the number of pixels in the hyperspectral image.

$$S = f^E (X) \tag{3}$$

where $X = (x_1, x_2, \ldots, x_N)$ is the input hyperspectral image. $f^E(\cdot)$ represents the operation of compressing $X$ into a low-dimensional representation by the encoder, which is highly consistent with the process of unmixing. Then, the decoder reconstructs $S$ back to the hyperspectral data

$$\hat{X} = f^D (S) \tag{4}$$

where $f^D(\cdot)$ decodes $S$ into the reconstructed image $\hat{X}$. In the context of LMM, this process is typically achieved by a fully connected layer, and the weights of this layer correspond to the endmember matrix of the hyperspectral image.

### B. GAT

GAT is an attention mechanism used in graph neural networks [42]. In the field of hyperspectral unmixing, a graph structure can be built by treating pixels or superpixels generated by segmenting the spatial region of hyperspectral images as nodes. Then, GAT aggregates and updates the information of the nodes on the graph. Compared with traditional GCNs, GAT can learn the importance weights of each node, instead of simply averaging the contribution of all neighboring nodes. Therefore, GAT can address the issue that aggregating neighboring node information equally may not be conducive to learning spatial information if two adjacent nodes have significantly different material compositions. More accurate spatial information can be learned by GAT. Moreover, GAT can handle graphs of arbitrary structures, making it suitable for any shape and size of the graph structure.

### C. PPNM

LMM cannot explain the nonlinear mixing residual contained in hyperspectral images [22]. Among the nonlinear mixture models, bilinear mixing models (BMMs) consider the main second-order scattering effects between objects by using the Hadamard product and nonlinear coefficients. So, BMMs commonly have relatively simple model forms to express nonlinearity. The PPNM [27] is a typical BMM that is usually used. It can be represented as follows:

$$x_j = g_b (M s_j) + \varepsilon_j = M s_j + b_j (M s_j) \odot (M s_j) + \varepsilon_j \tag{5}$$

where $g_b(\cdot)$ denotes the nonlinear mixing transformation, and $\odot$ represents the Hadamard product. The parameter $b_j$ is used to regulate the influence of the nonlinear components in a pixel, and all bilinear terms in a pixel are multiplied by the same parameter $b_j$. If $b_j = 0$, PPNM turns to the LMM. Since $b_j$ is closely related to the degree of the nonlinearity of pixels, it can be used to detect the degree of nonlinear mixing of the pixels.

## III. PROPOSED METHOD

This section introduces the proposed method GACAE which consists of two main modules: 1) a GAT-based spatial feature

extraction module; and 2) an improved autoencoder module. The overall structure of the GACAE algorithm is shown in Fig. 1. In brief, a hyperspectral image is first segmented into superpixels to build a graph structure. Graph attention convolution is then performed to make the obtained graph structure information of pixels. In the spectral dimension, the mapped graph structure information and the original hyperspectral image are concatenated into the AE. The encoder of the AE consists of four 2-D convolutional layers, and the decoder uses a PPNM layer to represent nonlinearity. A detailed description of the modules is provided as follows.

### A. GAT-Based Spatial Feature Extraction Module

Different land covers commonly present dominance in their own specific regions of the entire image scene and these regions are commonly irregular. The relationship between these irregular local regions can better reflect the land covers' global spatial distributions. To overcome the drawbacks of traditional convolution, a new module is designed to obtain the complex long-range spatial information of a hyperspectral image through the use of the GAT. It is noted that the construction of the GAT module relies on the definition of the graph nodes. Due to the high computational complexity of regarding every pixel directly as a node on the graph, a classical superpixel segmentation method simple linear iterative clustering (SLIC) [44] is employed to separate the hyperspectral data into a group of irregular homogeneous superpixel patches which are considered as graph nodes. And then, by establishing the adjacency relationship between each superpixel block, the hyperspectral image is converted into an undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ and $\mathcal{E}$ represent the sets of nodes and edges of the graph, respectively [45]. Inspired by [46], the proposed method regards the centroid of each superpixel as the nodes of the graph.

An incidence matrix $Q$ is generated which represents the mapping relationship between pixels and superpixels. Assuming the number of superpixel blocks is Z, and P denotes the superpixel set, the incidence matrix $Q \in \mathbb{R}^{N \times Z}$ can be defined as follows:

$$Q_{i,j} = \begin{cases} 1, & \text{if } X_{Fi} \in P_j \\ 0, & \text{otherwise} \end{cases}, X_F = Flatten(X) \tag{6}$$

where $Flatten(\cdot)$ denotes the operation of flattening the hyperspectral image by the spatial dimension. $X_{Fi}$ denotes the $i$th pixel in $X_F$, and $P_j$ is the $j$th superpixel.

Since superpixels can have different spatial and spectral relationships, an adjacency matrix $A \in \mathbb{R}^{Z \times Z}$ should be used to establish their adjacency relationship. Specifically, for any two superpixels, their adjacency relationship $A_{i,j}$ is defined as follows:

$$A_{i,j} = \begin{cases} 1, & \text{if } P_i \text{ and } P_j \text{ are adjacent } (i \neq j) \\ 0, & \text{otherwise} \end{cases}. \tag{7}$$

In addition, a node matrix $H = (h_1, h_2, \ldots, h_Z) \in \mathbb{R}^{B \times Z}$ is used to represent the information contained in the graph nodes of $\mathcal{V}$. $h_i$ denotes the information vector of the $i$th graph node. After $H$ and $A$ enter into the GAT layer, a weighted aggregated matrix $H' = (h'_1, h'_2, \ldots, h'_Z) \in \mathbb{R}^{B \times Z}$ of the neighboring
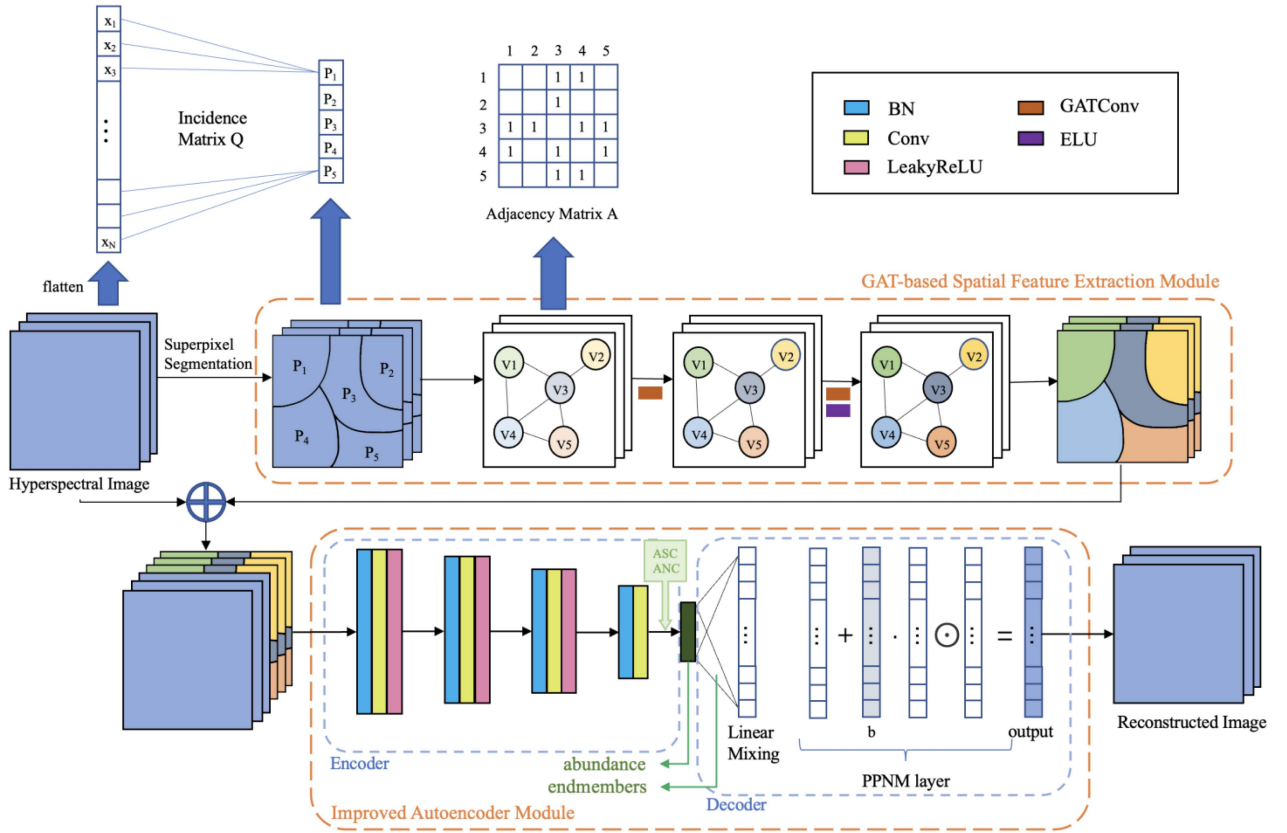
Fig. 1. Overall structure of the GACAE algorithm.

TABLE I
NETWORK LAYER STRUCTURE OF GACAE

| | | Layers | Activation function | Units |
|---|---|---|---|---|
| GAT Module | | Input | - | B |
| | | GATConv layer | - | 1/2B |
| | | GATConv layer | Elu | B |
| AE | | Input | - | 2B |
| | Encoder | BN + Conv2d(kernel =3×3, padding=1×1) | LeakyReLU | B |
| | | BN + Conv2d(kernel =3×3, padding=1×1) | LeakyReLU | 1/2B |
| | | BN + Conv2d(kernel =3×3, padding=1×1) | LeakyReLU | 1/4B |
| | | BN + Conv2d(kernel =1×1, padding=1×1) | - | R |
| | | abs + normalization | | |
| | Decoder | PPNM layer | - | B |

nodes can be obtained, the $i$th column vector of which can be denoted as follows:

$$h'_i = \sigma \left( \sum_{j \in \mathcal{N}_i} \alpha_{ij} W_1 h_j \right). \tag{8}$$

In (8), $\sigma(\cdot)$ represents the activation function, and $\mathcal{N}_i$ represents the set of adjacent nodes of the target node $i$. $W_1$ is a learnable weight matrix. The coefficient $\alpha_{ij}$ is the similarity weight between the target node $i$ and one of its adjacent nodes. $\alpha_{ij}$ can be calculated by the attention mechanism and its specific

expression is given as follows:

$$\alpha_{ij} = \text{softmax}(e_{ij}) = \frac{\exp(e_{ij})}{\sum_{k \in \mathcal{N}_i} \exp(e_{ik})} \tag{9}$$

$$e_{ij} = a^{\text{T}} \text{LeakyReLU}\left(W_2 \cdot [h_i \,\|\, h_j]\right) \tag{10}$$

where $e_{ij}$ represents the importance of node $j$ to node $i$. (10) is used [47] to avoid the collapse of $W_2$ and $a$ into a linear layer in the original GAT [42]. In the formula, $a$ and $W_2$ are the $2B \times 1$ learnable parameter vector and $2B \times 2B$ weight matrix, respectively, and $\|$ denotes the vector concatenation operation.

As shown in Fig. 1 and Table I, the constructed graph $\mathcal{G}$ is processed through two GAT layers and an ELU activation function. Compared to ReLU, the ELU activation function is

smoother, which helps to improve the robustness and stability of the network model [48]. To maintain the consistency of data structures, node features of the superpixel-based graph should be mapped to the corresponding pixel-level features through the corresponding relationship in $\boldsymbol{Q}$. In this sense, the pixels within the same superpixel could share the spatial relationship expressed by the corresponding graph node.

### B. Improved Autoencoder Module

*1) 2-D Convolution-Based Local Spatial Information Extraction:* The details of the autoencoder network architecture are shown in Table I. As a supplement to the GAT module, in the encoder part, 2-D convolutional layers with $3 \times 3$ kernels are used to learn short-range spatial information in smaller local regions. The batch normalization (BN) layer is added before each convolutional layer [45], [49], which is used to normalize each batch of data during the training. It prevents the network's activation values from being too biased toward a certain direction when the data distribution shifts, which can mitigate the gradient vanishing or exploding and make the training of the network more stable [50]. CNN is used to exploit the local spatial information of hyperspectral images to improve the accuracy of unmixing. This is because CNN can extract features within small local regions defined by the kernel size, allowing it to effectively learn the short-range local spatial characteristics of the hyperspectral image. Moreover, the LeakyReLU activation function is used after the CNN layer. LeakyReLU assigns a small gradient to each negative neuron value to replace a zero gradient, which allows negative values to backpropagate [51].

The features generated in the hidden layer are abundances, which should satisfy the abundance nonnegative constraint (ANC) and the abundance sum-to-one constraint (ASC)

$$ANC : \boldsymbol{s}_i \geq 0, \ \forall i \in \{1, 2, \dots, R\}$$

$$ASC : \sum_{i=1}^{R} \boldsymbol{s}_i = 1. \tag{11}$$

To satisfy (11), abundances are generated by normalizing the absolute values of the output of the encoder.

*2) PPNM Improved Decoder:* In the decoder, a PPNM layer is used to reconstruct the abundance features back to pixels, which is illustrated in Fig. 1. Specifically, a linear part is generated first using a fully connected layer [29], [32], and the weights of this linear layer are initialized by the endmembers obtained by vertex component analysis (VCA) algorithm [52]. The PPNM layer consists of a linear part and a nonlinear part, as shown in (5). According to the PPNM, the nonlinear part of a reconstructed pixel is obtained by taking the Hadamard product of its linear part, and the nonlinear parameter $\boldsymbol{b}$ is learned during the training.

### C. Loss Function

Further, the proposed method's loss function is

$$Loss = ReLoss + \lambda \cdot \|\boldsymbol{S}\|_{1/2} \tag{12}$$

where $ReLoss$ represents the reconstruction error which is calculated by the mean squared error (MSE).

$$ReLoss = \frac{1}{N} \sum_{i=1}^{N} (\hat{\boldsymbol{x}}_i - \boldsymbol{x}_i)^2 \tag{13}$$

where $\boldsymbol{x}_i$ and $\hat{\boldsymbol{x}}_i$ denote the input original pixel and the reconstructed pixel of the autoencoder, respectively.

The second term $\|\boldsymbol{S}\|_{1/2}$ in (12) is introduced to enforce the sparsity of the abundance [53]. Compared to the $L_0$ norm that leads to an intractable NP-hard problem and the $L_2$ norm that generates less sparse results, the $L_{1/2}$ norm can better enforce the sparsity of the abundance matrix [54]. Therefore, the intrinsic structural features of abundances including the spatial correlation and sparsity are promising to be effectively expressed in the unmixing results of the proposed method. The parameter $\lambda$ is used to control the degree of influence of the sparsity term.

## IV. EXPERIMENTAL RESULTS

### A. Data Description

Experiments with synthetic data and real hyperspectral remote sensing data[1] were carried out to evaluate the unmixing performance of the proposed method.

In terms of the generation of synthetic data, the reference endmembers were selected from the United States Geological Survey (USGS) spectral library. Each endmember has 224 bands in the spectral range from 0.39 to 2.56 $\mu$m. Abundances were generated by the Gaussian field method [32], and the maximum mixing degree was set to 0.8. Simulated data with a size of $64 \times 64$ pixels were generated based on the PPNM and Gaussian white noise was added. Different numbers of endmembers (from 3 to 5) and noise with different intensities (i.e., evaluated by the signal-to-noise ratio (SNR) [55] from 20 to 40 dB) were considered in the experiments.

For real hyperspectral data experiments, two subregions of the popularly used Samson and Jasper Ridge datasets were taken into account. It is noted that different from the synthetic data, true endmembers and abundances of the real data's ground truth are unknown. Therefore, similar to recently published works [17], [18], [33], [34], reference endmembers and abundances provided by specific processing methods in [56], [57], [58] were used for quantitative comparisons in this article.

The Samson dataset owned by Oregon state university was obtained by a push broom visible to near-infrared sensor [57]. It consists of $952 \times 952$ pixels each of which has 156 bands captured in the spectral range of 401 nm–889 nm with a spectral resolution up to 3.13 nm. For experimental evaluation, a $95 \times 95$-pixel subregion in Fig. 2 was used. Three typical endmembers including soil, trees, and water are commonly considered to exist in the observed area.

The Jasper Ridge dataset was captured by Airborne Visible-Infrared Imaging Spectrometer (AVIRIS) over the standard scene of the Jasper Ridge, a biological reserve in California [59]. It contains $512 \times 614$ pixels and each pixel was recorded at 224

---

[1][Online]. Available: https://rslab.ut.ac.ir/data.

Fig. 2.    Real hyperspectral remote sensing data: (a) Samson. (b) Jasper ridge.



Fig. 3.    Selection of the regularization parameter λ in GACAE.



Fig. 4.    Convergence curve of GACAE on the simulated data.

channels ranging from 0.38to 2.5 $\mu$m. The spectral resolution is up to 9.46 nm. A subregion containing $100 \times 100$ pixels with 198 bands (i.e., 1–3, 108–112, 154–166, and 220–224 bands were removed due to dense water vapor and atmospheric influence) in Fig. 2 was used in the experiments. Four main endmembers including trees, soil, water, and road exist in the region of interest.

### B. Quantitative Metric for Unmixing Evaluation

The root means squared error (RMSE) is used to evaluate the accuracy of abundance estimation, and the mean spectral angle distance (MSAD) is used to evaluate the accuracy of endmember extraction. Smaller values of RMSE in (14) and MSAD in (15) means better accuracy of abundance estimation and endmember extraction.

$$\text{RMSE} = \sqrt{\frac{1}{NR} \sum_{i=1}^{N} \|a_i - \hat{a}_i\|_2^2} \qquad (14)$$

$$\text{MSAD} = \frac{1}{R} \sum_{i=1}^{R} \cos^{-1}\left(\frac{m_i^{\text{T}} \hat{m}_i}{\| m_i \| \| \hat{m}_i \|}\right) \qquad (15)$$

where $\hat{a}_i$ is the estimated abundance vector of the $i$th pixel and $a_i$ is the corresponding reference of abundance. $m$ represents the reference endmember vector and $\hat{m}$ represents the estimated endmember vector. $x_i$ and $\hat{x}_i$ have the same definitions in (13).

### C. Compared Algorithms

The following typical algorithms are compared:
1) $L_{1/2}NMF$ [53]: An NMF-based linear unmixing algorithm that exploits the sparse characteristics of abundances in most real-world scenarios.
2) uDAS [60]: A cascaded autoencoder algorithm that combines both the edge-preserving denoising autoencoder and nonnegative sparse autoencoder for unmixing.
3) MAC_U [61]: A model-based unmixing algorithm that improves the AE's structure according to the nonlinear mixing models for nonlinear unmixing.
4) CyCU [18]: A linear unmixing method that uses the cyclic consistency of two autoencoders connected head to tail for unmixing.
5) rNMF [62]: A nonlinear unmixing method based on NMF that calculates the endmembers and abundances simultaneously using a block-coordinate descent algorithm.
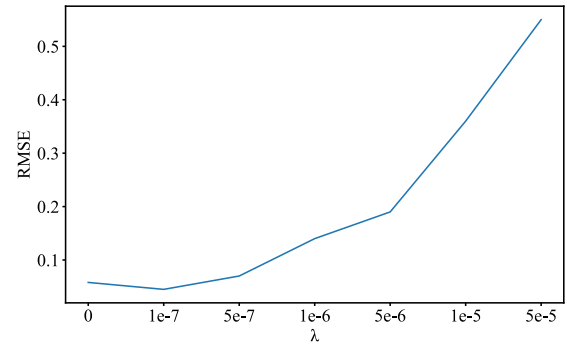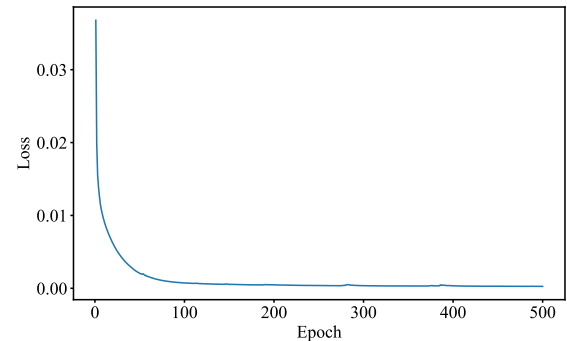
6) NAE [63]: An approach for blind nonlinear unmixing that uses a deep autoencoder network to solve the postnonlinear mixing problem.

The compared algorithms, excluding the rNMF, are the AE-based unmixing algorithms. In addition, MAC_U, rNMF, and NAE are specifically designed for nonlinear unmixing. All the algorithms are unsupervised unmixing algorithms. VCA is used to initialize endmembers. Hyperparameters including the number of epochs, learning rates, etc., were set according to the descriptions in the corresponding published papers. In quantitative comparison, every algorithm was independently run ten times and the mean and standard deviation of the MSAD values and the RMSE values were compared.

### D. Experiments With Synthetic Data

By using synthetic data constructed by four endmembers with SNR being 40 dB, the sparsity parameter λ in (12) was discussed in the first. It can be seen from Fig. 3 that the RMSE is the best when λ was set to 1e-7 which was used in the following experiments.

During the training, the network was optimized by the Adaptive Moment Estimation (Adam) optimizer. The learning rate for Adam and the linear layer of the decoder was set as 1e-3 and 1e-5, respectively. A weight decay rate of 1e-5 was used, and the number of epochs was set as 1000. Fig. 4 shows the convergence curve of GACAE under the condition that four endmembers were considered, the maximum abundance was set as 0.8, and SNR was set as 40 dB in simulated data. It can be observed that the

TABLE II
COMPARISON OF ESTIMATED ABUNDANCES' RMSES FOR THE SYNTHETIC DATA

| SNR | Number of endmembers | $L_{1/2}$NMF | uDAS | MAC_U | CyCU | rNMF | NAE | GACAE |
|---|---|---|---|---|---|---|---|---|
| 20dB | 3 | 0.0822±0.0079 | 0.1121±0.0097 | **0.0646±0.0048** | 0.1380±0.0127 | 0.0710±0.0108 | 0.0680±0.0046 | 0.0692±0.0083 |
| | 4 | 0.0786±0.0046 | 0.1164±0.0091 | 0.0714±0.0080 | 0.1391±0.0065 | 0.0716±0.0060 | 0.0682±0.0053 | **0.0671±0.0078** |
| | 5 | 0.0812±0.0098 | 0.1092±0.0138 | 0.0667±0.0109 | 0.1295±0.0084 | 0.0724±0.0162 | 0.0659±0.0104 | **0.0601±0.0105** |
| 30dB | 3 | 0.0701±0.0083 | 0.1007±0.0076 | **0.0587±0.0029** | 0.1163±0.0124 | 0.0693±0.0107 | 0.0593±0.0075 | 0.0602±0.0102 |
| | 4 | 0.0718±0.0073 | 0.1014±0.0064 | 0.0624±0.0043 | 0.1273±0.0106 | 0.0636±0.0099 | 0.0668±0.0064 | **0.0569±0.0088** |
| | 5 | 0.0759±0.0087 | 0.0981±0.0071 | 0.0568±0.0117 | 0.1170±0.0098 | 0.0633±0.0113 | 0.0610±0.0131 | **0.0565±0.0124** |
| 40dB | 3 | 0.0744±0.0041 | 0.1004±0.0072 | 0.0753±0.0130 | 0.1442±0.0225 | 0.0467±0.0088 | 0.0539±0.0129 | **0.0425±0.0102** |
| | 4 | 0.0691±0.0090 | 0.1285±0.0156 | 0.0564±0.0149 | 0.1065±0.0341 | 0.0462±0.0170 | 0.0491±0.0184 | **0.0328±0.0108** |
| | 5 | 0.0679±0.0109 | 0.1123±0.0199 | 0.0393±0.0138 | 0.1352±0.0245 | 0.0454±0.0124 | 0.0338±0.0134 | **0.0332±0.0091** |

TABLE III
COMPARISON OF EXTRACTED ENDMEMBERS' MSADS FOR THE SYNTHETIC DATA

| SNR | Number of endmembers | $L_{1/2}$NMF | uDAS | MAC_U | CyCU | rNMF | NAE | GACAE |
|---|---|---|---|---|---|---|---|---|
| 20dB | 3 | 0.0822±0.0079 | 0.1121±0.0097 | **0.0646±0.0048** | 0.1380±0.0127 | 0.0710±0.0108 | 0.0680±0.0046 | 0.0692±0.0083 |
| | 4 | 0.0786±0.0046 | 0.1164±0.0091 | 0.0714±0.0080 | 0.1391±0.0065 | 0.0716±0.0060 | 0.0682±0.0053 | **0.0671±0.0078** |
| | 5 | 0.0812±0.0098 | 0.1092±0.0138 | 0.0667±0.0109 | 0.1295±0.0084 | 0.0724±0.0162 | 0.0659±0.0104 | **0.0601±0.0105** |
| 30dB | 3 | 0.0701±0.0083 | 0.1007±0.0076 | **0.0587±0.0029** | 0.1163±0.0124 | 0.0693±0.0107 | 0.0593±0.0075 | 0.0602±0.0102 |
| | 4 | 0.0718±0.0073 | 0.1014±0.0064 | 0.0624±0.0043 | 0.1273±0.0106 | 0.0636±0.0099 | 0.0668±0.0064 | **0.0569±0.0088** |
| | 5 | 0.0759±0.0087 | 0.0981±0.0071 | 0.0568±0.0117 | 0.1170±0.0098 | 0.0633±0.0113 | 0.0610±0.0131 | **0.0565±0.0124** |
| 40dB | 3 | 0.0744±0.0041 | 0.1004±0.0072 | 0.0753±0.0130 | 0.1442±0.0225 | 0.0467±0.0088 | 0.0539±0.0129 | **0.0425±0.0102** |
| | 4 | 0.0691±0.0090 | 0.1285±0.0156 | 0.0564±0.0149 | 0.1065±0.0341 | 0.0462±0.0170 | 0.0491±0.0184 | **0.0328±0.0108** |
| | 5 | 0.0679±0.0109 | 0.1123±0.0199 | 0.0393±0.0138 | 0.1352±0.0245 | 0.0454±0.0124 | 0.0338±0.0134 | **0.0332±0.0091** |

Bold values are the best experimental results.

proposed method converges fast in the early stage and decreases slowly with the increase of epochs in the later stage to provide stable unmixing results.

Tables II and III provide the RMSE and MSAD of each algorithm for different simulated datasets. In Table II, it can be seen that the nonlinear unmixing algorithms including MAC_U, rNMF, and the proposed GACAE have better RMSE than most of the linear unmixing algorithms. As the number of endmembers and the SNR decreases, although the RMSE of GACAE increases slightly, it remains at a low level, indicating its robustness to noise and the number of endmembers. The reason may be that compared with the other algorithms, GACAE better uses the global and local spatial information of hyperspectral data in unmixing. Moreover, Table III also proves the superiority of GACAE. The MSAD of GACAE is competitive and can achieve the best performance in most cases, indicating that the improvement of abundance estimation by GACAE also helps to extract more accurate endmembers.

In addition, to further verify the beneficial function of different modules used in the proposed method, an ablation experiment was conducted. To generate the synthetic data, four endmembers were used and the SNR was 40 dB. In this experiment, the GAT module and the PPNM layer were removed to execute unmixing, respectively. Table IV compares the quantitative unmixing results obtained by using three module combinations. It is clear that the best results are achieved when all the modules are used in GACAE. Particularly, the GAT module has shown significant contribution to improving the unmixing accuracy even when the decoder is only based on the LMM. It implies that the complex spatial distribution of land covers can be effectively learned by the GAT module.

TABLE IV
ABLATION STUDY OF THE PROPOSED METHOD

| | RMSE | MSAD |
|---|---|---|
| GACAE | **0.0604±0.0108** | **0.0328±0.0108** |
| GACAE without GAT | 0.0696±0.0087 | 0.0348±0.0077 |
| GACAE without PPNM | 0.0802±0.0092 | 0.0395±0.0096 |

Bold values are the best experimental results.

TABLE V
RMSE AND MSAD OF THE SAMSON DATA

| | RMSE | MSAD |
|---|---|---|
| $L_{1/2}$NMF | **0.2237±0.0098** | 0.1344±0.0091 |
| uDAS | 0.2997±0.0106 | 0.0616±0.0057 |
| MAC_U | 0.2368±0.0078 | **0.0489±0.0074** |
| CyCU | 0.2319±0.0072 | 0.1534±0.0088 |
| rNMF | 0.2630±0.0089 | 0.2115±0.0093 |
| NAE | 0.2626±0.0102 | *0.0532±0.0089* |
| GACAE | *0.2254±0.0085* | 0.0548±0.0084 |

Bold values are the best experimental results.

### E. Experiments With Real Hyperspectral Data

For the Samson dataset, GACAE was trained with the following parameters: the learning rate for Adam was set as 5e-5, the learning rate for the decoder was 1e-4, and λ was 5e-5. The Samson dataset is trained for 500 epochs. Table V quantitatively lists the RMSEs of abundances and the MSADs of endmembers estimated by the compared algorithms for the Samson dataset. Fig. 5 displays the best abundance maps of each algorithm in
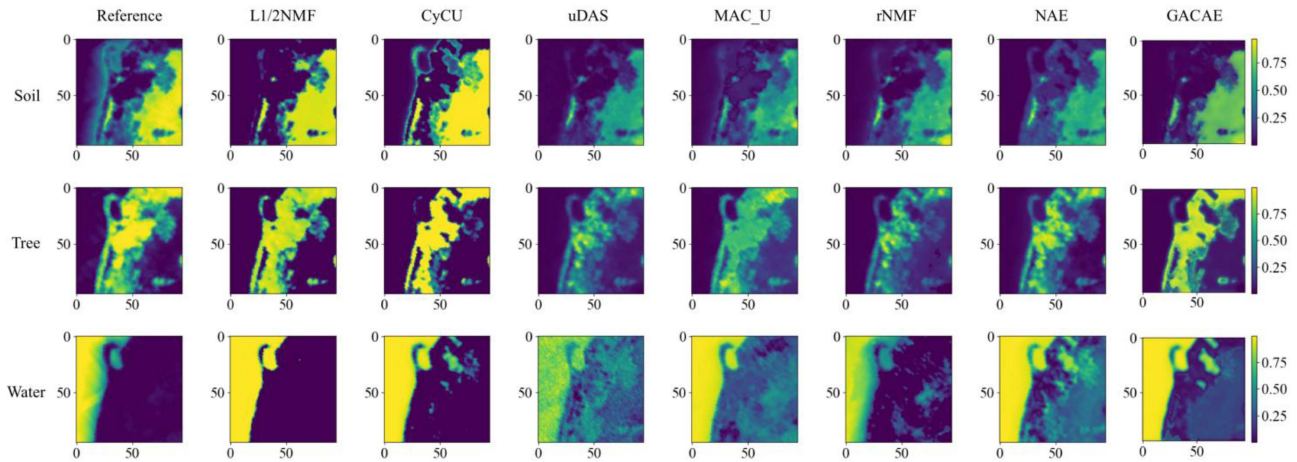
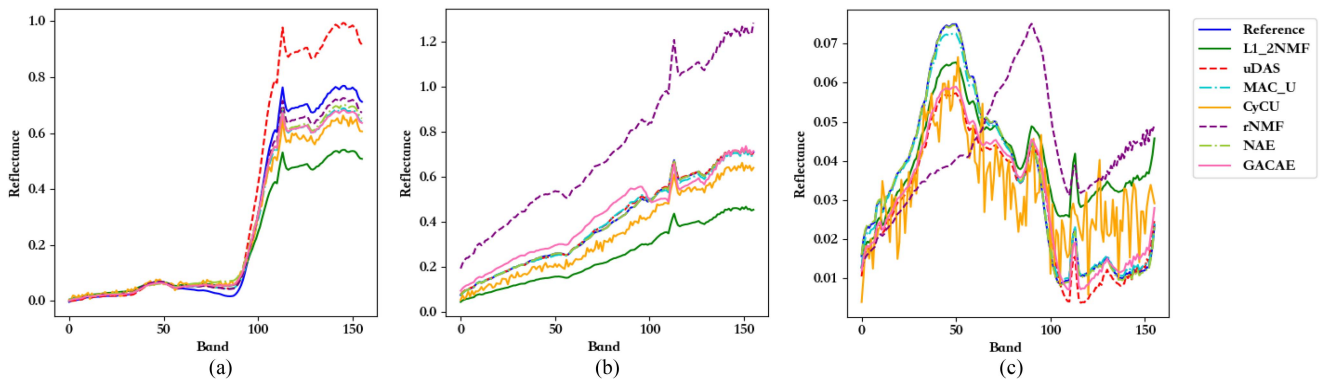Fig. 5.    Estimated abundance maps of the Samson data.



Fig. 6.    Estimated endmember spectral curves of samson data: (a) soil, (b) tree, (c) water.

ten experiments. The spectral curves of extracted endmembers can be clearly observed in Fig. 6.

From Figs. 5 and 6, it can be observed that the abundances of GACAE are sparser than that of the other nonlinear algorithms and similar to the reference abundance maps [58]. In comparison, GACAE's endmembers are much closer to the references. rNMF has a large deviation in its water endmember, and some oscillation exists in the CyCU's endmembers. In Table V, according to the reference endmembers [58], $L_{1/2}NMF$ achieves the smallest RMSE, but the MSAD of its endmembers is very large. In terms of MSAD of endmembers, two nonlinear unmixing algorithms MAC_U and NAE perform better, but they have poor abundances. In comparison, the unmixing results of GACAE are acceptable. The RMSE of GACAE's abundances is better than most nonlinear unmixing and AE-based unmixing methods, and its difference from the best is less than 5%. The MSAD of GACAE's endmembers is about 60% smaller than the $L_{1/2}NMF$ algorithm's.

For the Jasper dataset, the learning rate for Adam was 9e-3, the learning rate for the decoder was 5e-7, and λ was set as 2e-7. The Jasper dataset is trained for 1000 epochs. Figs. 7 and 8 show the qualitative unmixing results of the compared methods for the Jasper dataset. Table VI provides the quantitative

TABLE VI
RMSE AND MSAD OF THE JASPER RIDGE DATA

|  | RMSE | MSAD |
|---|---|---|
| $L_{1/2}$NMF | **0. 1218±0.0079** | **0.0947±0.0076** |
| uDAS | 0.1442±0.0081 | 0.1319±0.0087 |
| MAC_U | 0.1427±0.0077 | 0.1277±0.0068 |
| CyCU | 0.1231±0.0058 | 0.1283±0.0081 |
| rNMF | 0.1380±0.0062 | *0.1024±0.0103* |
| NAE | 0.1731±0.0113 | 0.1123±0.0120 |
| GACAE | *0.1226±0.0084* | 0.1118±0.0092 |

Bold values are the best experimental results.

comparison of the methods on the Jasper dataset. In terms of qualitative comparison, most methods can clearly reveal the ground covers' distribution in Fig. 7, but NAE may wrongly estimate the abundance map of the road. And it can be observed that the proposed method has better sparsity compared to the other nonlinear algorithms. From Fig. 8, it can be observed that most algorithms perform well in endmember extraction, but rNMF has significant differences in its tree endmember compared with
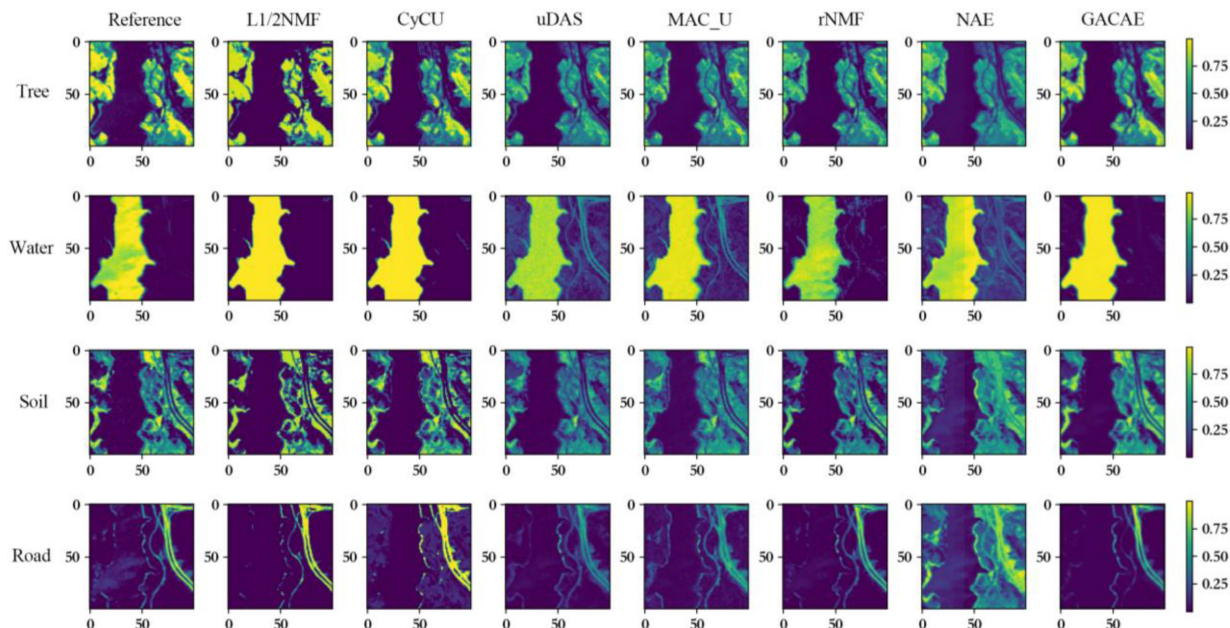
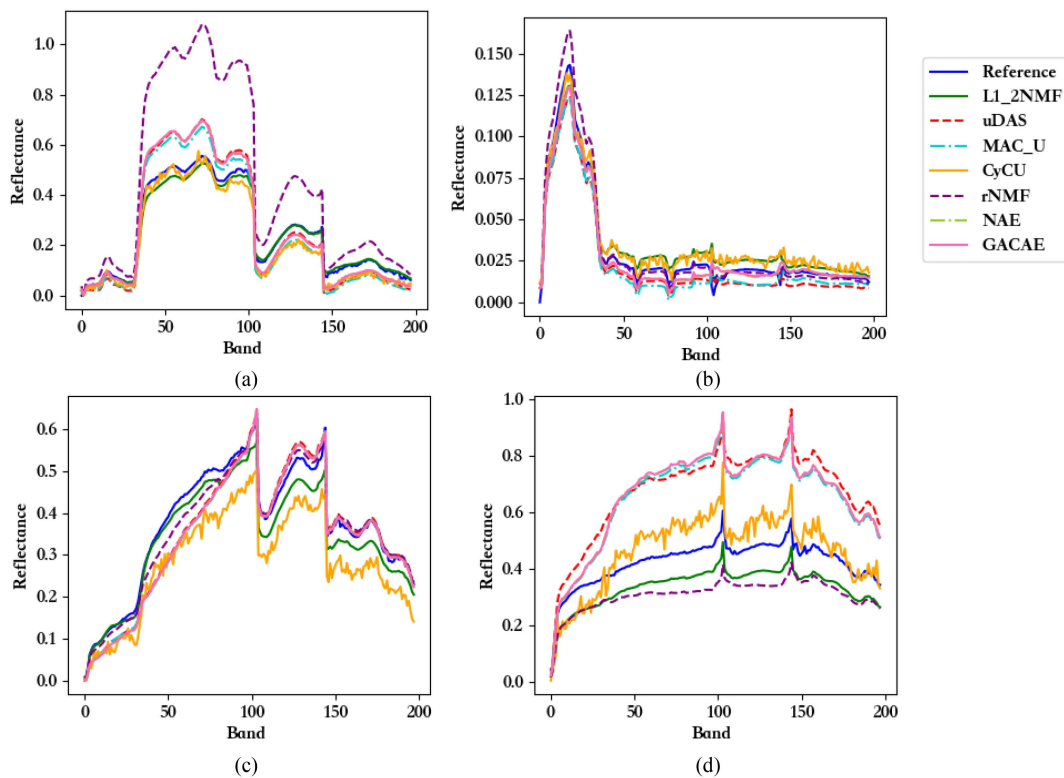Fig. 7. Estimated abundance maps of the Jasper ridge data.



Fig. 8. Estimated endmember curves of Jasper ridge data: (a) tree, (b) water, (c) soil, (d) road.

the reference. For the road endmember, uDAS, MAC_U, NAE, and GACAE all show significant differences from the reference endmembers. The two NMF-based methods extract better endmembers, indicating that deep learning-based methods may be further improved to perform better in endmember extraction.

In addition, the endmembers extracted by CyCU suffer from oscillation issues. In terms of quantitative comparison, according to Table VI, GACAE performs better than all the nonlinear unmixing algorithms which is similar to the case of the Samson dataset. It possibly indicates that the use of graph attention

TABLE VII
COMPUTATIONAL TIME OF EACH ALGORITHM ON DIFFERENT DATASETS (IN SECONDS)

| | $L_{1/2}$NMF | uDAS | MAC_U | CyCU | rNMF | NAE | GACAE |
|---|---|---|---|---|---|---|---|
| Synthetic Data ( 64×64 pixels) | 2.51 | 5.54 | 126.02 | 2.63 | 6.85 | 3.04 | 68.90 |
| Samson Dataset ( 95×95 pixels) | 2.24 | 4.78 | 141.64 | 4.41 | 5.16 | 3.62 | 80.20 |
| Jasper Ridge Dataset ( 100×100 pixels) | 3.52 | 6.61 | 154.20 | 4.02 | 7.18 | 4.50 | 242.46 |

convolutional operation can improve the nonlinear unmixing performance.

Finally, Table VII compares computational time on the synthetic dataset containing $64 \times 64$ pixels and two real hyperspectral remote sensing datasets. The $L_{1/2}NMF$ algorithm is the fastest, while the deep neural network-based algorithms usually need more time, especially for the data containing more pixels. An important work in the future is to improve the unmixing accuracy of deep neural network-based algorithms with a smaller epoch and less time.

## V. CONCLUSION

This article presents a graph attention convolution-based autoencoder method GACAE for nonlinear unmixing. It can effectively integrate long-range and local spatial information by using GAT and CNN. Therefore, the spatial information of hyperspectral images can be reasonably exploited to improve unmixing accuracy. Moreover, the decoder of the proposed method simulates the PPNM to give a physical interpretation for the nonlinear mixing effects so that the nonlinear unmixing problem can be well addressed. Experimental results on simulated and real hyperspectral remote sensing datasets demonstrate the effectiveness of GACAE and show its competitive performance compared to the state-of-the-art unmixing methods that more than ten percent of abundance estimation RMSE has been reduced by GACAE. In our future work, the proposed method's limitations such as the large time cost will be considered to be overcome.

## REFERENCES

[1] S. K. Roy, G. Krishna, S. R. Dubey, and B. B. Chaudhuri, "HybridSN: Exploring 3-D–2-D CNN feature hierarchy for hyperspectral image classification," *IEEE Geosci. Remote Sens. Lett.*, vol. 17, no. 2, pp. 277–281, Feb. 2020.

[2] A. D. Vibhute, S. V. Gaikwad, K. V. Kale, and A. V. Mane, "Hyperspectral image unmixing for land cover classification," in *Proc. IEEE India Council Int. Subsections Conf.*, 2021, pp. 1–5.

[3] D. Ö. Başkurt, F. Ömrüuzun, and Y. Y. Çetin, "Hyperspectral unmixing based analysis of forested areas," in *Proc. 23nd Signal Process. Commun. Appl. Conf.*, 2015, pp. 2329–2332.

[4] Y. Liang, P. P. Markopoulos, and E. S. Saber, "Subpixel target detection in hyperspectral images with local matched filtering in SLIC superpixels," in *Proc. 8th Workshop Hyperspectral Image Signal Process.: Evol. Remote Sens.*, 2016, pp. 1–5.

[5] D. Manolakis, C. Siracusa, and G. Shaw, "Hyperspectral subpixel target detection using the linear mixing model," *IEEE Trans. Geosci. Remote Sens.*, vol. 39, no. 7, pp. 1392–1409, Jul. 2001.

[6] J. Kaufman, J. J. Weinheimer, and M. Celenk, "Spatial-spectral feature extraction on hyperspectral imagery," in *Proc. 6th Workshop Hyperspectral Image Signal Process.: Evol. Remote Sens.*, 2014, pp. 1–4.

[7] J. M. Bioucas-Dias, A. Plaza, G. Camps-Valls, P. Scheunders, N. Nasrabadi, and J. Chanussot, "Hyperspectral remote sensing data analysis and future challenges," *IEEE Geosci. Remote Sens. Mag.*, vol. 1, no. 2, pp. 6–36, Jun. 2013.

[8] G. A. Shaw and H. K. Burke, "Spectral imaging for remote sensing," *Lincoln Lab. J.*, vol. 14, no. 1, pp. 3–28, Jan. 2003.

[9] L. Miao and H. Qi, "Endmember extraction from highly mixed data using minimum volume constrained nonnegative matrix factorization," *IEEE Trans. Geosci. Remote Sens.*, vol. 45, no. 3, pp. 765–777, Mar. 2007.

[10] R. Huang, X. Li, and L. Zhao, "Spectral–spatial robust nonnegative matrix factorization for hyperspectral unmixing," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 10, pp. 8235–8254, Oct. 2019.

[11] S. Zhang et al., "Spectral-spatial hyperspectral unmixing using nonnegative matrix factorization," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5505713.

[12] R. Guo, W. Wang, and H. Qi, "Hyperspectral image unmixing using autoencoder cascade," in *Proc. 7th Workshop Hyperspectral Image Signal Process.: Evol. Remote Sens.*, 2015, pp. 1–4.

[13] Y. Huang, J. Li, L. Qi, Y. Wang, and X. Gao, "Spatial-spectral autoencoder networks for hyperspectral unmixing," in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, 2020, pp. 2396–2399.

[14] B. Palsson, J. Sigurdsson, J. R. Sveinsson, and M. O. Ulfarsson, "Hyperspectral unmixing using a neural network autoencoder," *IEEE Access*, vol. 6, pp. 25646–25656, 2018.

[15] X. Tao et al., "A new deep convolutional network for effective hyperspectral unmixing," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 15, pp. 6999–7012, 2022.

[16] J. S. Bhatt and M. V. Joshi, "Deep learning in hyperspectral unmixing: A review," in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, 2020, pp. 2189–2192.

[17] Y. Su, J. Li, A. Plaza, A. Marinoni, P. Gamba, and S. Chakravortty, "DAEN: Deep autoencoder networks for hyperspectral unmixing," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 7, pp. 4309–4321, Jul. 2019.

[18] L. Gao, Z. Han, D. Hong, B. Zhang, and J. Chanussot, "CyCU-Net: Cycle-consistency unmixing network by learning cascaded autoencoders," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5503914.

[19] Y. Shan, S. Zhang, S. Hong, F. Li, C. Deng, and S. Wang, "Cascaded autoencoders for spectral-spatial remotely sensed hyperspectral imagery unmixing," in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, 2022, pp. 3271–3274.

[20] Z. Hua, X. Li, Q. Qiu, and L. Zhao, "Autoencoder network for hyperspectral unmixing with adaptive abundance smoothing," *IEEE Geosci. Remote Sens. Lett.*, vol. 18, no. 9, pp. 1640–1644, Sep. 2021.

[21] Z. Hua, X. Li, Y. Feng, and L. Zhao, "Dual branch autoencoder network for spectral-spatial hyperspectral unmixing," *IEEE Geosci. Remote Sens. Lett.*, vol. 19, 2022, Art. no. 5507305.

[22] R. Heylen, P. Scheunders, A. Rangarajan, and P. Gader, "Nonlinear unmixing by using different metrics in a linear unmixing chain," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 8, no. 6, pp. 2655–2664, Jun. 2015.

[23] J. Chen, M. Zhao, X. Wang, C. Richard, and S. Rahardja, "Integration of physics-based and data-driven models for hyperspectral image unmixing: A summary of current methods," *IEEE Signal Process. Mag.*, vol. 40, no. 2, pp. 61–74, Mar. 2023.

[24] B. Hapke, "Bidirectional reflectance spectroscopy: 1. Theory," *J. Geophys. Res.*, vol. 86, pp. 3039–3054, 1981.

[25] S. Jacquemoud and F. Baret, "PROSPECT: A model of leaf optical properties spectra," *Remote Sens. Environ.*, vol. 34, no. 2, pp. 75–91, Nov. 1990.

[26] Y. Altmann, N. Dobigeon, and J. Y. Tourneret, "Bilinear models for nonlinear unmixing of hyperspectral images," in *Proc. 3rd Workshop Hyperspectral Image Signal Process.: Evol. Remote Sens.*, 2011, pp. 1–4.

[27] Y. Altmann, N. Dobigeon, and J.-Y. Tourneret, "Unsupervised postnonlinear unmixing of hyperspectral images using a Hamiltonian Monte Carlo algorithm," *IEEE Trans. Image Process.*, vol. 23, no. 6, pp. 2663–2675, Jun. 2014.

[28] R. Heylen and P. Scheunders, "A multilinear mixing model for nonlinear spectral unmixing," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 1, pp. 240–251, Jan. 2016.

[29] K. T. Shahid and I. D. Schizas, "Unsupervised hyperspectral unmixing via nonlinear autoencoders," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5506513.

[30] B. Rasti and B. Koirala, "Blind nonlinear unmixing for intimate mixtures using Hapke model and CNN," in *Proc. 12th Workshop Hyperspectral Imag. Signal Process.: Evol. Remote Sens.*, 2022, pp. 1–5.

[31] J. Chen, C. Richard, and P. Honeine, "Nonlinear unmixing of hyperspectral data based on a linear-mixture/nonlinear-fluctuation model," *IEEE Trans. Signal Process.*, vol. 61, no. 2, pp. 480–492, Jan. 2013.

[32] M. Zhao, M. Wang, J. Chen, and S. Rahardja, "Hyperspectral unmixing for additive nonlinear models with a 3-D-CNN autoencoder network," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5509415.

[33] S. Shi, L. Zhang, Y. Altmann, and J. Chen, "Deep generative model for spatial–spectral unmixing with multiple endmember priors," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5527214.

[34] B. Palsson, M. O. Ulfarsson, and J. R. Sveinsson, "Convolutional autoencoder for spectral–spatial hyperspectral unmixing," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 1, pp. 535–549, Jan. 2021.

[35] X. Zhang, Y. Sun, J. Zhang, P. Wu, and L. Jiao, "Hyperspectral unmixing via deep convolutional neural networks," *IEEE Geosci. Remote Sens. Lett.*, vol. 15, no. 11, pp. 1755–1759, Nov. 2018.

[36] Y. Cai, Z. Zhang, Z. Cai, X. Liu, and X. Jiang, "Hypergraph-structured autoencoder for unsupervised and semisupervised classification of hyperspectral image," *IEEE Geosci. Remote Sens. Lett.*, vol. 19, 2022, Art. no. 5503505.

[37] J. Qu, Y. Xu, W. Dong, Y. Li, and Q. Du, "Dual-branch difference amplification graph convolutional network for hyperspectral image change detection," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5519912.

[38] Y. Ding, J. Feng, Y. Chong, S. Pan, and X. Sun, "Adaptive sampling toward a dynamic graph convolutional network for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5524117.

[39] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," in *Proc. Int. Conf. Learn. Representations*, 2017, pp. 1–14.

[40] A. Sha, B. Wang, X. Wu, and L. Zhang, "Semisupervised classification for hyperspectral images using graph attention networks," *IEEE Geosci. Remote Sens. Lett.*, vol. 18, no. 1, pp. 157–161, Jan. 2021.

[41] D. Hong, L. Gao, J. Yao, B. Zhang, A. J. Plaza, and J. Chanussot, "Graph convolutional networks for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 7, pp. 5966–5978, Jul. 2021.

[42] P. Velickovic, G. Cucurull, A. Casanova, A. Romero, P. Lio, and Y. Bengio, "Graph attention networks," in *Proc. Int. Conf. Learn. Representations*, 2018, pp. 1–12.

[43] J. Lai, X. Wang, Q. Xiang, Y. Song, and W. Quan, "Review on autoencoder and its applications," *J. Commun.*, vol. 42, no. 9, pp. 218–230, Sep. 2021.

[44] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Süsstrunk, "SLIC superpixels compared to state-of-the-art superpixel methods," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 11, pp. 2274–2282, Nov. 2012.

[45] Q. Liu, L. Xiao, J. Yang, and Z. Wei, "CNN-enhanced graph convolutional network with pixel- and superpixel-level feature fusion for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 10, pp. 8657–8671, Oct. 2021.

[46] S. Wan, C. Gong, P. Zhong, B. Du, L. Zhang, and J. Yang, "Multiscale dynamic graph convolutional network for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 5, pp. 3162–3177, May 2020.

[47] S. Brody, U. Alon, and E. Yahav, "How attentive are graph attention networks?," in *Proc. Int. Conf. Learn. Representations*, 2018, pp. 1–24.

[48] D. Clevert, T. Unterthiner, and S. Hochreiter, "Fast and accurate deep network learning by exponential linear units (ELUs)," in *Proc. Int. Conf. Learn. Representations*, 2016, pp. 1–14.

[49] H. Zhang, J. Zou, and L. Zhang, "EMS-GCN: An end-to-end mixhop superpixel-based graph convolutional network for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5526116.

[50] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *Proc. Int. Conf. Mach. Learn.*, 2015, pp. 1–11.

[51] A. L. Maas, A. Y. Hannun, and A. Y. Ng, "Rectifier nonlinearities improve neural network acoustic models," *Proc. Int. Conf. Mach. Learn.*, vol. 30, no. 1, p. 3, 2013.

[52] J. M. P. Nascimento and J. M. B. Dias, "Vertex component analysis: A fast algorithm to unmix hyperspectral data," *IEEE Trans. Geosci. Remote Sens.*, vol. 43, no. 4, pp. 898–910, Apr. 2005.

[53] Y. Qian, S. Jia, J. Zhou, and A. Robles-Kelly, "Hyperspectral unmixing via L1/2 sparsity-constrained nonnegative matrix factorization," *IEEE Trans. Geosci. Remote Sens.*, vol. 49, no. 11, pp. 4282–4297, Nov. 2011.

[54] M. W. Berry, M. Browne, A. N. Langville, V. P. Pauca, and R. J. Plemmons, "Algorithms and applications for approximate nonnegative matrix factorization," *Comput. Statist. Data Anal.*, vol. 52, no. 1, pp. 155–173, Sep. 2007.

[55] "Signal-to-noise ratio (SNR) in hyperspectral imagers," in Tech Briefs, Sep. 2020. [Online]. Available: https://www.techbriefs.com/component/content/article/tb/supplements/pit/features/articles/37674

[56] Samson and Jasper Ridge Dataset. Accessed: Jul. 2022. [Online]. Available: https://rslab.ut.ac.ir/data

[57] F. Zhu, Y. Wang, B. Fan, S. Xiang, G. Meng, and C. Pan, "Spectral unmixing via data-guided sparsity," *IEEE Trans. Image Process.*, vol. 23, no. 12, pp. 5412–5427, Dec. 2014.

[58] F. Zhu, "Hyperspectral unmixing: Ground truth labeling, datasets, benchmark performances and survey," 2017, *arXiv:1708.05125*.

[59] A. M. Ahmed, O. Duran, Y. Zweiri, and M. Smith, "Hybrid spectral unmixing: Using artificial neural networks for linear/non-linear switching," *Remote Sens.*, vol. 9, no. 8, 2017, Art. no. 775.

[60] Y. Qu and H. Qi, "uDAS: An untied denoising autoencoder with sparsity for spectral unmixing," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 3, pp. 1698–1712, Mar. 2019.

[61] H. Li, R. A. Borsoi, T. Imbiriba, P. Closas, J. C. M. Bermudez, and D. Erdoğmuş, "Model-based deep autoencoder networks for nonlinear hyperspectral unmixing," *IEEE Geosci. Remote Sens. Lett.*, vol. 19, 2022, Art. no. 5506105.

[62] C. Févotte and N. Dobigeon, "Nonlinear hyperspectral unmixing with robust nonnegative matrix factorization," *IEEE Trans. Image Process.*, vol. 24, no. 12, pp. 4810–4819, Dec. 2015.

[63] M. Wang, M. Zhao, J. Chen, and S. Rahardja, "Nonlinear unmixing of hyperspectral data via deep autoencoder networks," *IEEE Geosci. Remote Sens. Lett.*, vol. 16, no. 9, pp. 1467–1471, Sep. 2019.

**Danni Jin** received the B.S. degree in Computer Science and Technology from the School of Information Engineering, Hangzhou University of Electronic Science and Technology, Hangzhou, China, in 2020. She is currently working toward the M.S. degree in computer science and technology from the School of Computer Science and Technology, Donghua University, Shanghai, China.

Her research interests include hyperspectral unmixing, pattern recognition, and deep neural network.

**Bin Yang** (Member, IEEE) received the B.S. degree in geographic information system from Huazhong Agricultural University, Wuhan, China, in 2011, the M.S. degree in cartography and geographic information system from South China Normal University, Guangzhou, China, in 2015, and the Ph.D. degree in circuit and system from Fudan University, Shanghai, China, in 2019.

Since July 2019, he has been a Lecturer with the School of Computer Science and Technology, Donghua University, Shanghai, China. His main research interests include hyperspectral remote sensing image analysis, machine learning, pattern recognition, and computational intelligence.