Hyperspecral Unmixing based on Multilinear Mixing Model using Convolutional Autoencoders

Tingting Fang, Fei Zhu, *Member, IEEE*, and Jie Chen, *Senior Member, IEEE*

***Abstract*—Unsupervised spectral unmixing consists of representing each observed pixel as a combination of several pure materials known as endmembers, along with their corresponding abundance fractions. Beyond the linear assumption, various nonlinear unmixing models have been proposed, with the associated optimization problems solved either by traditional optimization algorithms or deep learning techniques. Current deep learning-based nonlinear unmixing mainly focuses on additive, bilinear-based formulations. The multilinear mixing model (MLM) offers a unique perspective by interpreting the reflection process by discrete Markov chains, allowing it to account for interactions between endmembers up to infinite order. However, explicitly simulating the physics of MLM using neural networks has remained a challenging problem. In this paper, we propose a novel autoencoder-based network for unsupervised unmixing based on MLM. Leveraging an elaborate network design, this approach explicitly models the relationships among all model parameters: endmembers, abundances, and transition probability. The network operates in two modes: MLM-1DAE, which considers only pixel-wise spectral information, and MLM-3DAE, which explores spectral-spatial correlations within input patches. Experiments on both the synthetic and real datasets validate the effectiveness of the proposed method, demonstrating competitive performance compared to classic MLM-based solutions. The code is available at https://github.com/ting-Fang09/Hyperspectral-unmixing-MLM-AE.**

***Index Terms*—Spectral unmixing (SU), multilinear mixed model (MLM), convolutional autoencoders, deep neural network.**

I. INTRODUCTION

SPECTRAL unmixing (SU) has been a hot topic in many fields related to remote sensing, *e.g.*, mineral exploration and agriculture over the past decades [1]. A hyperspectral image is a three-dimensional data cube, with hundreds of continuous spectral bands captured across a certain wavelength range over the same geographic target. Due to the relatively low spatial resolution, each spectra pixel in the image is supposed to be a mixture of several pure material signatures, namely *endmembers*. SU refers to identifying the endmembers and to estimating the corresponding fractions for each pixel, namely *abundances* [1].

Different assumptions about the physical propagation process of photons lead to the categorization of unmixing models into linear-based and nonlinear-based ones [2]. Among them, the linear mixing model (LMM) is the most prevalent due to its simplicity in both the mixing process and unmixing strategy design. It assumes that a photon interacts with only one material before reaching the sensors, resulting in each observed spectrum being a linear combination of the endmembers [3].

However, severe nonlinear effects occur in many scenarios, thus requiring more sophisticated unmixing models [2]. Several nonlinear, physics-driven unmixing methods have been devised to explicitly explain the mixing process. They are the Hapke's model for intimate unmixing process [4], the bilinear-based methods [5]–[7], and the multilinear-based methods (which are the topic of this paper) [8]–[11]. The bilinear mixing models hypothesize that each photon interacts with two endmember materials in series, thus formulated as the LMM superimposed by a second-order term. Representative bilinear-based unmixing models, such as Fan model [5], generalized bilinear model [6] and polynomial postnonlinear mixing model (PPNMM) [7], introduce various constraints on model parameters. More recently, the bilinear models were extended to incorporate automatic shadow compensation [12], or solved by nonnegative tensor factorization [13].

In contrast to the data-driven SU methods, *e.g.* the kernel-based ones [14]–[16], the aforementioned physics-driven nonlinear unmixing models provide enhanced interpretability by explicitly accounting for the interactions of light with endmembers before reaching the sensor.

The physics-driven multilinear mixing model (MLM) has been proposed, for the first time, to consider all degrees of endmember interactions [8]. Essentially, the MLM extends the PPNMM by modeling the second-order nonlinear effects to the infinite order. In this model, the reflection process that a single light undergoes before reaching the observer is elegantly represented using a discrete Markov chain. Notably, this model employs only a single pixel-wise transition probability parameter, which characterizes the possibility of light undergoing further interactions to effectively quantify nonlinear intensity. This inherent simplicity of the MLM, combined with its explicit interpretation in physics, renders it computationally tractable and effective in real-world scenarios [9], [10]. The original MLM is established in a supervised manner, assuming prior knowledge of the endmembers. Subsequently, improvements have been made to MLM from different aspects. In [9], the so-called MLMp is formulated in an unsupervised manner, which employs a simplified optimization function to mitigate the parameter under-estimation issue and to reduce the algorithm complexity. The band-wise nonlinear unmixing algorithm (BNLSU) [10] characterizes the transition probability parameter by a pixel-wise vector defined over the wavelength range instead of using a single scalar. In [11],

The work was supported by the National Natural Science Foundation of China under Grant 61701337. *(Corresponding author: Fei Zhu.)*

T. Fang and F. Zhu are with the Center for Applied Mathematics, Tianjin University, Tianjin, China (Emails: ting_fang; fei.zhu@tju.edu.cn).

J. Chen is with the School of Marine Science and Technology, Northwestern Polytechnical University, Xi'an, China (Email: dr.jie.chen@ieee.org).

a graph regularized MLM is proposed, where the underlying data manifold structure is incorporated by the Laplacian graph regularizers. [17] enhances the robustness of the BNLSU with a $l_{2,1}$-norm-based loss function. By far, MLM and its variants have been tackled using classical optimization methods, such as Alternating Direction Method of Multipliers (ADMM). In contrast, the primary motivation driving our research is rooted in the endeavor to faithfully emulate the unmixing process inherent in the MLM using deep learning techniques.

Taking advantage of the progress in deep learning, the autoencoders (AEs) and their variants have been successfully applied to SU [18]. The AEs are with an encoder-decoder architecture, in which neural networks are employed to find the low-dimensional representation of the input data by minimizing the reconstruction error [19]. The architecture of the AEs inherently offers a flexible unsuperivsed unmixing framework, namely the encoder converts the input spectra to the corresponding abundance vectors, *i.e.*, the outputs of the hidden layer, and the decoder reconstructs the input from the compressed representations, with the weights in the last linear layer interpreted as the endmember matrix. The majority of the AE-based unmixing networks are based on the LMM assumption [20]–[23]. To mitigate the effects of noise in hyperspectral data, robust unmixing methods were developed, including the stacked nonnegative sparse autoencoders [24] and the part-based denoising autoencoder with $l_{2,1}$ norm and denoising constraints [25]. The uDAS [26] improves the part-based autoencoders by untying the encoder and decoder. The adversarial autoencoder in [27] enhances the model robustness to outliers and noise by using adversarial training and adding abundance prior.

Regularization terms were also integrated to the loss function of AEs from different aspects, *e.g.*, the minimum simplex volume penalty to incorporate the geometry among data points [24], [28], [29], and the sparsity augmentation term on the abundance vectors in the so-called EndNet [30]. The global-local smooth autoencoder explores the local homogeneity and global self-similarity of the hyperspectral image [31]. In [20], a CNN-based unmixing network (CNNAEU) is proposed to capture the spectral-spatial correlations within the hyperspecral data.

Besides AEs, nonnegative matrix factorization (NMF) [32] has traditionally played a pivotal role in unsupervised SU [33]. It achieves unmixing by factorizing the hyperspectral matrix into the product of an unknown nonnegative abundance matrix and an endmember matrix. More recently, the deep unrolling network-based SU have emerged [34]–[36]. These networks emulate the optimization steps in NMF by assigning each iteration to a separate layer and stacking multiple such layers to construct a hierarchical network. Compared to AE-based methods, deep unrolling-based approaches demonstrate accelerated convergence in blind unmixing [34]. Additionally, the associated networks offer good interpretability as it thoroughly unfolds NMF. Nonetheless, a limitation lies in their challenging adaptation to mimic physics-driven nonlinear SU models since NMF inherently operates in a linear fashion.

Addressing the nonlinear SU in the context of deep learning is a challenging problem. It is because that nonlinear SU models usually require explicit descriptions of photons' interactions with the endmembers, while the black box characteristics of neural networks may lead to unclear mixing process [18]. A category of nonlinear SU methods by deep learning is data-driven, and the nonlinearity is generally formulated in the form of implicit functions. In [37], a deep autoencoder network is designed to imitate an implicit nonlinear transform to the linear mixture, and the nonlinear layer is added to the decoder accordingly. Other existing nonlinear SU models are realized by augmenting LMM with an additive nonlinear mixture part, where the nonlinearity is modeled implicitly [38], [39]. Different networks are applied, *i.e.*, a 3D-CNN structure to capture the spectral-spatial information [38], and an LSTM to enhance the spectral correlations [39]. Attempts have also been made to realize the physics-driven unmixing models explicitly. Considering the bilinear mixing models, two deep autoencoders within a multi-task learning framework are combined, but with the extra computational burden [40]. In a recent study [41], a versatile unmixing scheme was designed to handle a range of nonlinear models, including bilinear-based and finite-degree interaction models. Two autoencoder networks were established for Fan model [5] and PPNMM. For Fan model, they introduced a cross-product layer to capture endmember interactions and explored incorporating higher-degree cross-product terms by stacking them beneath the second-order terms. However, this approach, despite its computational complexity, had limitations, as it could only handle endmember interactions up to a finite order, making it unsuitable for modeling MLM. Additionally, the study treated pixel-wise nonlinear coefficients as learnable network parameters rather than features, which limited adaptability for adjusting parameter quantities or shapes during training based on varying input data characteristics.

Replicating the physics-driven MLM in blind unmixing by deep learning is more challenging when compared to additive, bilinear-based unmixing models. In this paper, we propose an unsupervised AE-based unmixing network for the MLM mechanism. Our choice of AEs is motivated by two primary considerations. Firstly, AEs are well-suited for unsupervised unmixing tasks, demonstrating the capability to simultaneously estimate both endmembers and abundances. Secondly, AE architectures offer a high degree of flexibility and adaptability, allowing us to carefully design network structures that effectively replicate the nonlinear mixing mechanisms inherent in MLM. The main contributions are summarized as follows.

1) This work pioneers an AE-based unmixing network specifically tailored to the physics-driven MLM. Our focus is on achieving explicit unmixing in accordance with the MLM mechanism, primarily realized through decoder design. The proposed methods faithfully reflect the MLM's mixing formula and loss function. The network incorporates an interpretable design, particularly in the decoder, by implementing operations on learned features and specific layers to explicitly model the mathematical connections between endmembers, abundances, and transition probabilities.
2) The proposed network achieves blind unmixing by con-

currently estimating all model parameters in the MLM. Notably, we learn both abundance vector and transition probability as pixel-wise features, instead of network parameters, enhancing network adaptability across diverse datasets. Moreover, it achieves unsupervised unmixing while preserving the intricate complexity of the underlying MLM model, as opposed to the simplified optimization function employed in MLMp [9].

3) The proposed unmixing network offers versatility by two distinct modes: MLM-1DAE and MLM-3DAE. The first network exclusively considers spectral correlations, faithfully simulating the original MLM. The MLM-3DAE incorporates spectral-spatial information through 3D convolutions on image patches, enhancing unmixing performance in scenarios where spatial context is pivotal.

The remainder of the paper is organized as follows. Section II revisits the relevant unmixing models. In Section III, the proposed autoencoder-based unmixing network for MLM is presented. Experimental results and analysis are given in Section IV. Finally, Section V concludes the paper.

II. PRELIMINARY

Consider a hyperspectral image containing N pixels, with $\boldsymbol{E} = [\boldsymbol{e}_1, \boldsymbol{e}_2, ..., \boldsymbol{e}_R] \in \mathbb{R}^{B\times R}$ being the underlying endmember matrix composed by R signatures over B spectral bands. Let $\boldsymbol{x} \in \mathbb{R}^{B\times 1}$ be an observed pixel, $\boldsymbol{a} \in \mathbb{R}^{R\times 1}$ be the associated abundance vector of the R endmembers, and $\varepsilon \in \mathbb{R}^{B\times 1}$ is the additive Gaussian noise. The LMM considers to represent each pixel as a linear combination of the endmembers, and is formulated by

$$\begin{aligned} &\boldsymbol{x} = \boldsymbol{E}\boldsymbol{a} + \varepsilon, \\ &s.t.\ a_i \geq 0,\ \forall i = 1, 2, ..., R \\ &\quad \mathbf{1}_R^\top \boldsymbol{a} = 1, \end{aligned} \tag{1}$$

where the non-negative constraint (ANC) and abundance sum-to-one constraint (ASC) are usually imposed considering the physical meaning.

A. Multilinear Mixing Model (MLM) [8]

In [8], the MLM extends the bilinear-based unmixing models to include all orders of interactions among endmembers. In this context, a discrete Markov chain is employed to model the reflection process of a single light ray before reaching the observer. Note that relevant concept of Dirichlet-Markov random process has been applied in multi-temporal spectral unmixing task, where the state variables represent the time-varying abundance coefficients or endmembers [42], [43]. The MLM assumes the following:

- A photon will interact with at least one endmember.
- After each interaction with an endmember, the photon has a probability P for further interactions, and a probability $(1 - P)$ of escaping the scenario and being received by the sensors.
- The probability of interacting with the i-th endmember is proportional to its corresponding abundance fraction.
- When scattered by some endmember, the light intensity will change according to that material's albedo $\boldsymbol{w} \in [0, 1]^B$. By neglecting the nuance between the albedo and the endmember spectrum, the former is approximated by the latter with $\boldsymbol{w}_i \approx \boldsymbol{e}_i$, $\forall i = 1, 2, ..., R$.

Based on the above assumptions, the MLM is expressed by

$$\begin{aligned} \boldsymbol{x} =& (1-P)\sum_{i=1}^{R}\boldsymbol{e}_i a_i + (1-P)P\sum_{i=1}^{R}\sum_{k=1}^{R}(\boldsymbol{e}_i \odot \boldsymbol{e}_k)a_i a_k \\ &+ (1-P)P^2\sum_{i=1}^{R}\sum_{k=1}^{R}\sum_{l=1}^{R}(\boldsymbol{e}_i \odot \boldsymbol{e}_k \odot \boldsymbol{e}_l)a_i a_k a_l + \ldots + \varepsilon \\ =& (1-P)\boldsymbol{y} + P\boldsymbol{y} \odot \boldsymbol{x} + \varepsilon, \end{aligned} \tag{2}$$

where $\boldsymbol{y} = \boldsymbol{E}\boldsymbol{a}$ represents the linear part with LMM, and $\odot$ is the Hadamard element-wise product. Here, P is the transition probability for further interactions, with its value restricted to $P < 1$. When $P = 0$, the model (2) will degrade to the LMM. In theory, the above model is also well-defined for $P < 0$, although the physical reasoning no longer holds with negative probability [8]. From (2), it is straightforward that

$$\boldsymbol{x} = \frac{(1-P)\boldsymbol{y}}{1-P\boldsymbol{y}} + \varepsilon. \tag{3}$$

As a result, the following optimization problem is considered,

$$\begin{aligned} &\arg\min_{\boldsymbol{a},P} \left\| \boldsymbol{x} - \frac{(1-P)\boldsymbol{y}}{1-P\boldsymbol{y}} \right\|_2^2 \\ &s.t. \quad \boldsymbol{a} \geq 0 \quad \text{and} \quad \mathbf{1}_R^\top \boldsymbol{a} = 1 \\ &\qquad P < 1, \end{aligned} \tag{4}$$

where the reconstruction error between the observed pixel and the reconstructed one is minimized. The unmixing problem in (4) is formulated in a supervised manner, namely, the endmembers are supposed to be extracted in prior using some endmemeber extraction strategy, such as the vertex component analysis (VCA) [44].

B. Unsupervised MLM (MLMp) [9]

Later in [9], the authors proposed to address the MLM in (2) as an unsupervised unmixing problem, where the endmember matrix, the abundance vector, and the transition probability parameter were estimated collaboratively. To this end, the reconstruction errors over all the pixels are added up, yielding

$$\begin{aligned} &\arg\min_{\boldsymbol{E},\{\boldsymbol{a}_j,P_j\}_{j=1}^N} \sum_{j=1}^{N} \|(1-P_j)\boldsymbol{y}_j + P_j\boldsymbol{y}_j \odot \boldsymbol{x}_j - \boldsymbol{x}_j\|_2^2 \\ &s.t. \quad \boldsymbol{a}_j \geq 0 \quad \text{and} \quad \mathbf{1}_R^\top \boldsymbol{a}_j = 1 \\ &\qquad P_j \leq 1, \quad \forall j = 1, 2, ..., N \\ &\qquad 0 \leq \boldsymbol{E} \leq 1. \end{aligned} \tag{5}$$

To achieve the unsupervised unmixing with modest computational complexity, the above optimization simplifies the original problem (4) by eliminating the denominator and applying a block coordinate descent method for problem-solving.

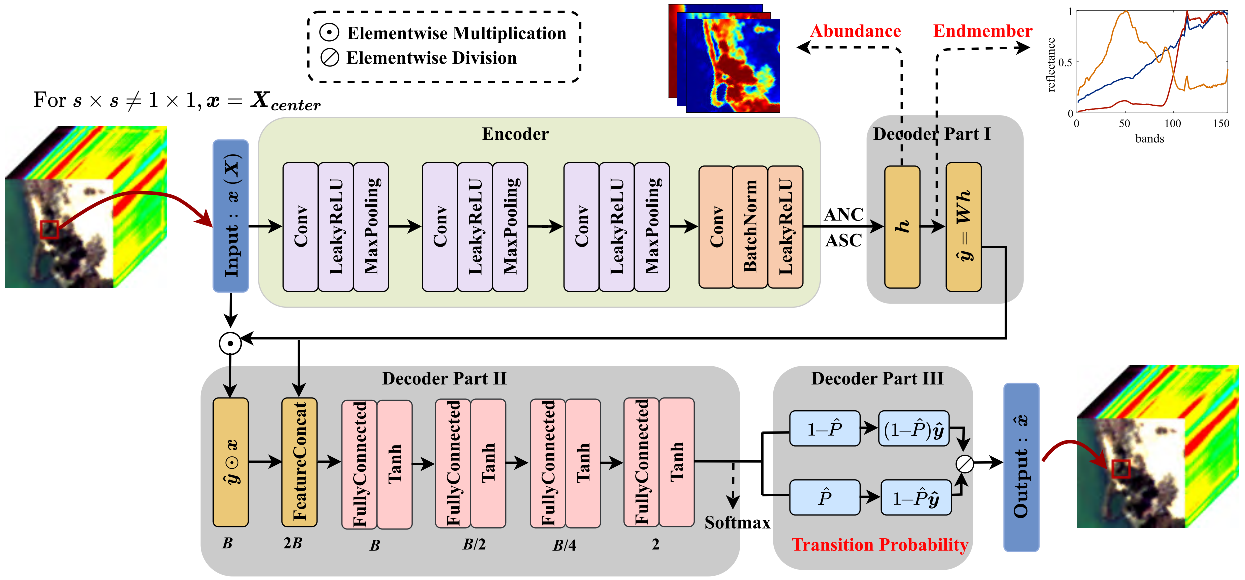


Fig. 1. Schematic of the proposed autoencoder network for MLM-based unsupervised unmixing. The encoder compresses the observed pixel into the abundance vector. The decoder is composed by three parts. Decoder Part I represents the linear part, with the network weights interpreted as the endmember matrix. Decoder Part II estimates the pixel-wise transition probability by concatenating and transforming two relevant features. Decoder Part III reconstructs the pixel using model parameters and learned features, adhering to MLM model.

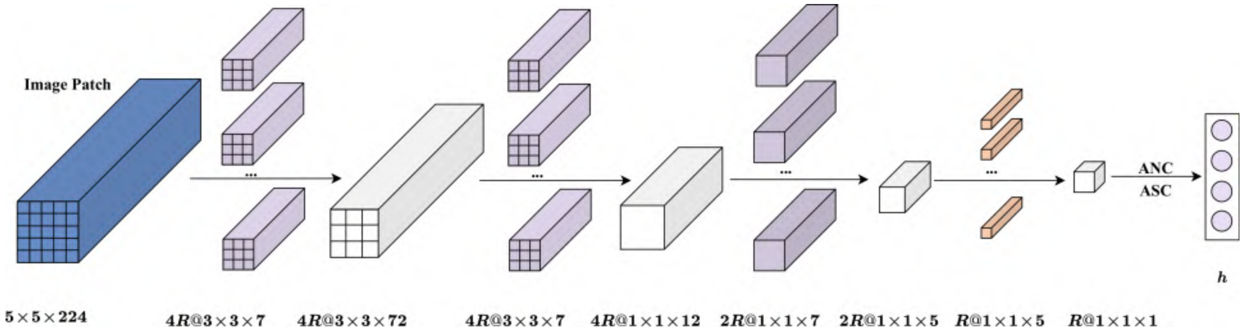


Fig. 2. Illustration of feature changes during the encoding process in MLM-3DAE unmixing network, with $s = 5, B = 224$.

III. PROPOSED METHOD

In this section, we propose to address the MLM-based unmixing model in (3) in an unsupervised manner, by exploiting an AE-based network. The proposed network explicitly simulates the MLM mechanism and is capable to estimate the endmembers, abundance vectors, and pixel-wise, transition probability parameters jointly.

A. Network overview

As illustrated in Fig. 1, the proposed unmixing network adopts an encoder-decoder architecture. In both MLM-1DAE and MLM-3DAE modes, the primary function of the encoder network is to compress the input pixel data into a lower-dimensional representation, yielding the abundance vector.

In contrast, the decoder network exhibits greater complexity and comprises three distinct components. The decoder network is designed to reconstruct each observed pixel according to (4), while avoiding introducing unnecessary network parameters. Decoder Part I handles the LMM using a single linear layer, where the network weights are interpreted as the endmember matrix. Decoder Part II is responsible for determining pixel-wise transition probabilities P, which are constrained to the interval [0, 1]. To this end, two relevant features influencing P are concatenated, transformed, and passed through a softmax layer, yielding a probability score vector in the form of $[P, 1-P]$. Decoder Part III is designed to reconstruct the pixel using network weights and learned features, adhering to the MLM model defined in (3). This process incorporates operations such as feature reuse and concatenation.

Regarding training, the network minimizes the average distance between input pixels and their reconstructions. The corresponding loss function is in alignment with the original MLM optimization problem described in (4), with the key distinction of operating in an unsupervised mode.

B. Encoder

The encoder network $f(\cdot)$, which is shown in the upper part of Fig. 1, compresses the input data into the R-dimensional representation. The number of neurons in the encoder bottleneck layer is determined based on the number of endmembers or virtual dimensionality of the hyperspectral dataset being analyzed. In this study, we assume prior knowledge of the endmember count R, which can be estimated by the unsupervised hyperspectral dimensionality reduction methods [45], [46], such as HySime [47]. This assumption is based on the low-rank nature of hyperspectral data, leading us to consider $R < B$. Given the pixel $\boldsymbol{x}$, the corresponding output of the encoder is taken as the estimated abundance vector $\hat{\boldsymbol{a}}$, namely

$$\text{Abundance Estimation : } \boldsymbol{a} \leftarrow \boldsymbol{h} = f(\boldsymbol{x}) \tag{6}$$

To enforce both the ANC and ASC constraints, the output feature of the encoder is normalized as follows

$$h_i = \text{Softmax}(h_i) = \frac{e^{h_i}}{\sum_{j=1}^{R} e^{h_j}}, \tag{7}$$

where h_i denotes the i-th entry of the estimated abundance vector $\boldsymbol{h}$.

According to whether considering the spatial information, we employ two encoders with slight differences, resulting in MLM-1DAE and MLM-3DAE. In general, the encoder plays a crucial role in acquiring the prior spectral or spectral-spatial information regarding the correlation between abundances and observed pixels. Its primary function is to estimate and regularize abundance. Consequently, the encoder exhibits flexibility, accommodating a diverse choices of structures [48], ranging from more complex designs (such as Transformer-based networks [49]) to simpler alternatives.

1) MLM-1DAE: The first encoder takes the pixel spectrum $\boldsymbol{x}$ as input, and 1D convolutions operations are performed across the spectral dimension in the subsequence blocks. Unmixing with this encoder is consistent with the original MLM [8], where only the spectral information is exploited during the unmixing process.

To be precise, the first layer inputs a pixel of size $1 \times 1 \times B$. Three identical blocks, each including a Convolution layer, a LeakyRelu layer and a MaxPooling layer are followed. The fourth block has a convolution layer, a BatchNorm layer and a LeakyRelu layer. Regarding the Convolution layers, the number of feature maps are $4R$, $4R$, $2R$, and R, and the kernel sizes are 7, 7, 7, and 5, respectively, where no padding is added and all strides are set to 1. For the MaxPooling layer, the kernel size is 3 and the stride is 3, with no padding. We refer to the unmixing network using the encoder with 1D convolutions as MLM-1DAE.

2) MLM-3DAE: To explore the rich spectral-spatial information of the hyperspectral imagery, we design a second encoder and refer to the unmixing network with it as MLM-3DAE. This encoder has a similar network structure as in MLM-1DAE and mainly utilizes the 3D convolutional layers to capture the spectral-spatial dependencies. In MLM-3DAE, the input comprises an image patch $\boldsymbol{X}$ of size $s \times s \times B$, centered at pixel $\boldsymbol{x}$. The aim is to extract the feature vector of size $1 \times 1 \times R$ from $\boldsymbol{X}$, to form the abundance vector for the neighborhood center $\boldsymbol{x}$. To achieve spatial dimension downsampling from $s \times s$ to 1×1, we set the convolution kernel size in corresponding dimensions by $\max(3, odd(\lceil s/3 \rceil)) \times \max(3, odd(\lceil s/3 \rceil))$ without padding. Additionally, through the use of Convolution and MaxPooling layers over the spectral dimension, this dimension is decreased from B to 1. Subsequently, R feature maps are concatenated to generate the abundance vector.

The integration of 3D convolutional layers within MLM-3DAE equips our model with the ability to capture the intricate spectral-spatial relationships inherent in hyperspectral data, contributing to its enhanced unmixing performance. For illustration purposes, we provide an example with $s = 5$ and $B = 224$ in Fig. 2. In the first block, the convolutional layer comprises $4R$ feature maps with 3D convolutional kernels of size $3 \times 3 \times 7$, employing no padding and a stride of $1 \times 1 \times 1$. The MaxPooling layer uses a $1 \times 1 \times 3$ kernel with no padding and a stride of $1 \times 1 \times 3$. The settings for the second block remain unchanged from the first block. Following this block, the spatial dimensions are reduced to 1×1. The third block has $2R$ feature maps with 3D convolution kernels of size $1 \times 1 \times 7$, no padding and stride size $1 \times 1 \times 1$; the kernel size in MaxPooling layer is $1 \times 1 \times 3$. Finally, the fourth block produces R feature maps, corresponding to the number of endmembers, with 3D convolutional kernels of size $1 \times 1 \times 5$ and no padding.

C. Decoder

As depicted in Fig. 1, the decoder consists of three distinct components designed to reconstruct each pixel following the MLM mechanism in (3). The encoded feature dimension undergoes a mapping from R to B to reconstruct the input. It is important to note that the complexity of decoder is inherent from the complexity of the MLM itself. The optimization problem within MLM for unsupervised unmixing is ill-posed and non-convex, involving the decoupling of various parameters, namely endmembers, abundances, and transition parameters, all subject to various constraints. In the following, we provide a detailed explanation of each decoder part.

1) Decoder Part I: The first layer of the decoder is the fully connected (FC) layer without bias and activation function, following the last layer of the encoder. It processes the linear transformation on the encoder outputs, *i.e.*, abundance vectors, and is taken as the linear mixture of the LMM model. The network weights $\boldsymbol{W}$ of the FC layer is interpreted as the endmember matrix by

$$\text{Endmember Extraction: } \boldsymbol{E} \leftarrow \boldsymbol{W}.$$

Here, the layer weights are restricted to the interval [0,1] by applying $\phi(x) = \min(1, \max(0, x))$, considering the physical meaning of endmembers. In practice, we initialize the weights of this FC layer using the endmembers extracted by the VCA technique [44].

2) Decoder Part II: Within the MLM framework, as given in (2), an observed pixel can be regarded as a combination of two fundamental components: a linear component $\boldsymbol{y}$ and a nonlinear part $\boldsymbol{y} \odot \boldsymbol{x}$. The balance between these components is adjusted by the transition probability parameter P.

Derived from this MLM formulation, the design of Decoder Part II takes shape. Given that both of these components are influenced by the value of P, we choose to concatenate two relevant features from the preceding layer. These features consist of the linear component, denoted as $\hat{\boldsymbol{y}} = \boldsymbol{W}\boldsymbol{h}$, and the element-wise product of this linear component with the input data, represented as $\hat{\boldsymbol{y}} \odot \boldsymbol{x}$. This concatenation results in a feature representation with an enlarged dimensionality, specifically $2B$. Subsequently, this augmented feature is processed through a sequence of blocks comprising FC layers, interleaved with hyperbolic tangent (Tanh) activation functions. These blocks work together to progressively reduce the dimensionality of the feature from $2B$ to 2, aiming to extract high-level information pertaining to the transition probability parameter from the interconnected features.

3) Decoder Part III: The last part of the decoder estimates the transition probability parameter by a softmax function, and reconstructs the spectrum according to (3) for MLM model. The softmax function is applied to the output of the precedent layer, denoted by $\boldsymbol{z} = [z_1, z_2]^\top$, and transforms it to the binomial probability distribution with $\boldsymbol{P} = [1 - \hat{P}, \hat{P}]^\top$. As a result, the transition probability is estimated by

$$\text{Probability Estimation}: \ P \leftarrow \hat{P} = \frac{\exp z_2}{\sum_{i=1}^{2} \exp z_i} \tag{8}$$

In this manner, the estimated value of P naturally falls within the interval $[0, 1]$, a suitable range for characterizing it as the probability of a photon undergoing further interactions in the reflectance process of the MLM model. It is noteworthy that we maintain the constraint of P within the interval [0, 1], aligning with its interpretation as a probability in the MLM. This restriction ensures straightforward and physically meaningful outcomes, distinguishing it from [8], where the theoretical possibility of $P < 0$ is acknowledged but not incorporated into our current model.

By reusing the acquired features and the network parameters, the rest part of Decoder III reconstructs the input data $\boldsymbol{x}$ through an the element-wise division operation between the features $(1 - \hat{P})\hat{\boldsymbol{y}}$ and $(1 - \hat{P}\hat{\boldsymbol{y}})$, adhering to the MLM formula presented in (3).

D. Loss Function

In the proposed network, we adopt the commonly-used mean square error (MSE) between the observed pixels and reconstructed ones as the loss function, which is defined as follows:

$$J_{\text{MSE}}(\boldsymbol{x}, \hat{\boldsymbol{x}}) = \|\boldsymbol{x} - \hat{\boldsymbol{x}}\|_2^2, \tag{9}$$

where $\hat{\boldsymbol{x}} = \frac{(1-\hat{P})\hat{\boldsymbol{y}}}{1-\hat{P}\hat{\boldsymbol{y}}}$ is the reconstructed pixel obtained from the output of the network. The loss function becomes

$$\mathcal{L} = \frac{1}{K} \sum_{k=1}^{K} J_{\text{MSE}}(\boldsymbol{x}_i, \hat{\boldsymbol{x}}_i), \tag{10}$$

where K is the batch size.

Notice we directly address a nearly identical optimization problem as in MLM, as given in (4), while in an unsupervised unmixing setting. This differs from the MLMp algorithm, which uses a modified loss function to eliminate the denominator, aiming to mitigate high nonlinearity in parameter estimation and to reduce optimization complexity, as given in (5), for unsupervised unmixing.

IV. EXPERIMENTS

In this section, we prove the effectiveness of the proposed multilinear-based unmixing networks, MLM-1DAE and MLM-3DAE, through comprehensive comparative studies conducted on synthetic images and two real hyperspectral images. To this end, eight state-of-the-art unmixing methods were selected for comparison accounting for different aspects. Regarding traditional unmixing schemes, we have incorporated the fully constrained least squares (FCLS) [50], the polynomial postnonlinear mixing model (PPNMM) [7], and two multilinear-based methods, *i.e.* the supervised multilinear mixture model (MLM) [8] and the unsupervised multilinear unmixing (MLMp) [9]. For the deep learning-based unmixing approaches, we chose the sparse autoencoder network for endmember extraction and unmixing (EndNet) [30], the convolutional neural network autoencoder unmixing (CNNAEU) [20], the additive nonlinear unmixing with a 3D-CNN autoencoder (3D-NAE) [38], and the autoencoder for postpolynomial nonlinear unmixing (PPNMM-AE) [41].

For fairness, VCA [44] is applied in prior for endmember extraction in all the supervised unmixing methods, namely FCLS, PPNMM and MLM. The same VCA results are also applied to initialize the endmembers in all the unsupervised unmixing methods, namely MLMp, EndNet, CNNAEU, 3D-NAE, PPNMM-AE and the proposed methods. Note that in CNNAEU, we selected the version CNNAEU2 over CNNAEU1 for enhanced abundance estimation. In PPNMM-AE, we employed VCA instead of the original K-means clustering for endmember initialization.

The endmember estimation is evaluated by using the averaged spectral angle distance (SAD) of all the endmembers, which is defined by

$$\text{SAD}_{\text{end}} = \frac{1}{R} \sum_{i=1}^{R} \arccos\left(\frac{\boldsymbol{m}_i^\top \hat{\boldsymbol{m}}_i}{\|\boldsymbol{m}_i\|_2 \, \|\hat{\boldsymbol{m}}_i\|_2}\right), \tag{11}$$

where $\boldsymbol{m}_i$ and $\hat{\boldsymbol{m}}_i$ represent the i-th groundtruth and estimated endmembers, respectively.

The abundance estimation is evaluated using the root mean square error (RMSE), given by

$$\text{RMSE}_{\text{abun}} = \sqrt{\frac{1}{NR} \sum_{i=1}^{N} \|\boldsymbol{a}_i - \hat{\boldsymbol{a}}_i\|_2^2} \tag{12}$$

where $\boldsymbol{a}_i$ and $\hat{\boldsymbol{a}}_i$ are the groundturth and estimated abundance vectors for the i-th pixel.

The transition probabilities obtained by the multilinear-based unmixing methods are compared by the RMSE, with

$$\text{RMSE}_{\text{P}} = \sqrt{\frac{1}{N} \sum_{i=1}^{N} \left\| P_i - \hat{P}_i \right\|^2} \tag{13}$$

where P_i and $\hat{P}_i$ represent the groundtruth and the estimated values, respectively.

On real images, as no ground-truth information is available for evaluating the estimated endmembers, abundances and transition probabilities, the pixel SAD is used to assess the unmixing performance, defined by

$$\mathrm{SAD} = \frac{1}{N}\sum_{i=1}^{N} \arccos\left(\frac{\boldsymbol{x}_i^\top \hat{\boldsymbol{x}}_i}{\|\boldsymbol{x}_i\|_2 \, \|\hat{\boldsymbol{x}}_i\|_2}\right), \tag{14}$$

where $\boldsymbol{x}_i$ and $\hat{\boldsymbol{x}}_i$ are the actual and reconstructed pixels. Note that while pixel SAD serves as an evaluation metric, it may not necessarily reflect the accuracy of parameter estimation performance.

A. Experiments with Synthetic Data

1) Data Generation: A series of synthetic datasets of size 256×256 was generated, taking into account three crucial aspects: the number of endmembers, noise levels, and mixing mechanisms. We selected $R = 4$ and $R = 8$ endmembers from the United States Geological Survey (USGS) digital spectral library composed by spectra measured over 224 consecutive bands. The abundance maps were generated using the Hyperspectral Imagery Synthesis (HYDRA) toolbox [1]. We introduced various noise levels by adding Gaussian white noise with signal-to-noise ratios (SNR) of 20dB and 30dB, respectively. The synthetic images were generated according to three distinct mixing mechanisms: linear-based LMM in (1), bilinear-based PPNMM [7], and multilinear-based MLM in (3). The PPNMM is defined by

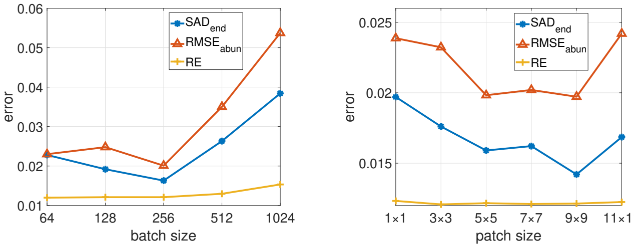
$$\boldsymbol{x} = \boldsymbol{E}\boldsymbol{a} + \gamma(\boldsymbol{E}\boldsymbol{a}) \odot (\boldsymbol{E}\boldsymbol{a}) + \varepsilon, \tag{15}$$

where $\gamma \in [-0.3, 0.3]$ was drawn from a uniform distribution, following the original paper [7]. In MLM, the pixel-wise transition probabilities were randomly generated from a half-normal distribution with $\sigma = 0.3$, and the values of P larger than 1 are set to zero, as did in [8] [9] [2].

2) Parameter settings: In this series of experiments, the hyperparameters in the proposed MLM-1DAE and MLM-3DAE are set as follows.

We employ the Adam optimizer, utilizing varying learning rate strategies for different parts of the network. Notably, we adopt a relatively smaller learning rate for the linear decoder part to mitigate training instability, as did in [39]. Specifically, for Decoder Part I, we set learning rates to 10^{-7} in all cases, except when dealing with MLM-generated data at SNR=30, R=4. In this case, the learning rate for Decoder Part I is slightly adjusted to 5×10^{-7}. As for the remaining components of the unmixing networks, we consistently use a learning rate of 10^{-4} for R=4 and 5×10^{-4} for R=8. It is noteworthy that the learning rates may display minor fluctuations when applied to distinct datasets. The number of training epochs is set to 300.

We tune the batch-size from the candidate value set $[64, 128, 256, 512, 1024]$ using MLM-1DAE, on the synthetic image with SNR=30, R=4 generated using the MLM model. Fig. 3(a) reports the resulting errors with respect to three monitoring metrics, namely average endmember SAD, abundance RMSE, and pixel RE. Accordingly, we choose the batch size to be 256 in both MLM-1DAE and MLM-3DAE.



(a) Impact of batch size (b) Impact of patch size

Fig. 3. Changes of average endmember SAD, abundance RMSE, and pixel RE along with (a) varying batch size using MLM-1DAE; (b) different input patch size using MLM-3DAE.

Regarding MLM-3DAE, the patch size $s \times s$ for the input subcube is selected from the candidate set $[1\times1, 3\times3, 5\times5, 7\times7, 9\times9, 11\times11]$, and the corresponding changes of the average endmember SAD, abundance RMSE, and pixel RE are given in Fig. 3(b). Here, we apply the MLM-1DAE with 1D convolution in $s \times s = 1 \times 1$. As observed, modest patch sizes with 5×5, 7×7 and 9×9 can efficiently use the spatial information of neighboring pixels by 3D convolution operations, yielding better unmixing performance when compared to merely using the spectral information with 1×1. However, a greater spatial patch size with 11×11 will not only increase the computing overhead, but also deteriorate the unmixing performance. It is because of the overfitting issue caused by an increasing number of network parameters. Considering that increasing the patch size leads to higher computational complexity, we maintain a fixed patch size of 5×5 in MLM-3DAE for subsequent experiments.

3) Results Analysis: All the experiments are repeated ten times. Comparisons of the unmixing performance, including mean and deviation, across different unmixing methods, are presented in TABLE I for R=4 and in TABLE II for R=8, where best and the second-best results are marked by a circled number. Unmixing performance is notably influenced by the compatibility between the data mixing mechanism and the applied unmixing model, particularly evident when the dataset involves a smaller number of endmembers with R=4.

For endmember extraction, VCA proves highly effective by providing small SAD of endmembers, thus serving as a good endmember initialization, especially for unmixing linear or bilinear-generated data. However, when faced with MLM-generated data, improved accuracy in endmember estimation is achieved by other nonlinear unmixing techniques. Specifically, on MLM-generated data with varying noise levels and endmember counts, MLM-1DAE and MLM-3DAE consistently provide the two best endmember estimation results. There is one exception in the case of SNR=20 and R=8, where MLM-1DAE falls slightly short, with 3D-NAE performing best in endmember estimation. The classic MLMp performs well in endmember estimation when the data is relatively clean with SNR=30 and when the model aligns well with the data. However, for noisy cases with SNR=20, MLMp struggles to accurately extracting the endmembers. Fig. 4 shows the estimated endmembers of the best performing run of all the comparing methods on MLM-generated data at SNR=30, R=4.

---

[1] http://www.ehu.es/ccwintco/index.php/Hyperspectral Imagery Synthesis tools for MATLAB

[2] For comparison fairness, the P-value in MLMp is constrained to the range [0,1] in the experiments with synthetic datasets.

TABLE I

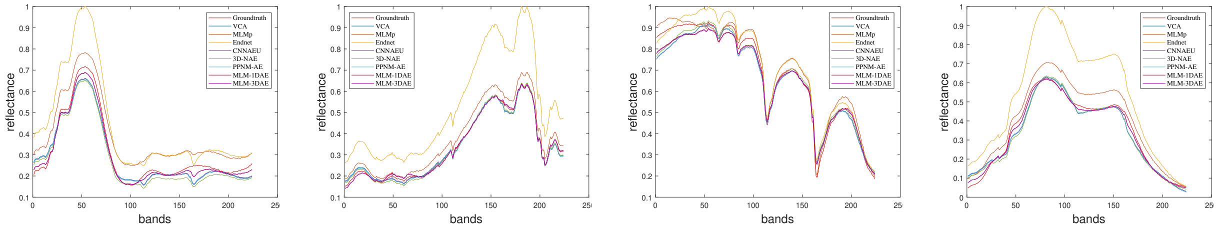
Comparison of unmixing results $\times 10^{-2}$ (mean ± standard deviation) on synthetic data with R=4, SNR=20dB and 30dB, generated by LMM, PPNMM and MLM models, averaged over 10 runs

Mix. / Unmix.		SNR=20 dB			SNR=30 dB		
		LMM	PPNMM	MLM	LMM	PPNMM	MLM
SAD_{end}	VCA	① 1.01 ± 0.00	① 1.29 ± 0.00	5.29 ± 0.00	① 0.26 ± 0.00	① 0.28 ± 0.00	6.06 ± 0.00
	MLMp	33.55 ± 0.00	33.83 ± 0.00	30.48 ± 0.00	11.13 ± 0.00	11.28 ± 0.00	4.90 ± 0.00
	EndNet	1.32 ± 0.01	1.48 ± 0.01	5.00 ± 0.01	0.62 ± 0.01	0.73 ± 0.01	4.72 ± 0.01
	CNNAEU	1.04 ± 0.01	1.32 ± 0.01	5.16 ± 0.01	0.42 ± 0.02	0.48 ± 0.02	5.65 ± 0.01
	3D-NAE	1.38 ± 0.37	1.92 ± 0.41	6.02 ± 0.05	0.72 ± 0.31	0.77 ± 0.36	6.04 ± 0.02
	PPNMM-AE	② 1.02 ± 0.00	① 1.29 ± 0.00	5.32 ± 0.00	② 0.27 ± 0.00	② 0.30 ± 0.00	5.77 ± 0.00
	MLM-1DAE	1.07 ± 0.04	1.32 ± 0.05	① 4.89 ± 0.01	0.37 ± 0.09	0.40 ± 0.04	① 2.91 ± 0.03
	MLM-3DAE	1.09 ± 0.01	1.38 ± 0.05	② 4.90 ± 0.02	0.40 ± 0.06	0.53 ± 0.09	② 2.93 ± 0.02
$\text{RMSE}_{\text{abun}}$	FCLS	① 1.49 ± 0.00	① 1.73 ± 0.00	9.52 ± 0.00	① 0.45 ± 0.00	① 0.52 ± 0.00	9.24 ± 0.00
	MLM	1.81 ± 0.00	2.01 ± 0.00	5.83 ± 0.00	0.56 ± 0.00	0.61 ± 0.00	7.30 ± 0.00
	MLMp	2.69 ± 0.00	3.03 ± 0.00	5.33 ± 0.00	1.12 ± 0.00	1.21 ± 0.00	5.22 ± 0.00
	PPNMM	1.77 ± 0.00	1.99 ± 0.00	8.37 ± 0.00	② 0.55 ± 0.00	② 0.60 ± 0.00	10.81 ± 0.00
	EndNet	3.32 ± 0.03	3.28 ± 0.02	① 4.78 ± 0.09	3.15 ± 0.02	3.13 ± 0.01	7.56 ± 0.06
	CNNAEU	2.08 ± 0.02	2.35 ± 0.02	6.79 ± 0.04	1.71 ± 0.01	1.77 ± 0.01	8.59 ± 0.02
	3D-NAE	4.04 ± 1.03	4.02 ± 1.37	13.89 ± 0.13	2.86 ± 1.18	2.18 ± 0.91	14.66 ± 0.15
	PPNMM-AE	2.02 ± 0.00	2.13 ± 0.00	9.33 ± 0.00	1.65 ± 0.00	1.66 ± 0.00	9.61 ± 0.00
	MLM-1DAE	1.70 ± 0.04	② 1.96 ± 0.16	5.39 ± 0.20	1.11 ± 0.19	1.12 ± 0.05	② 3.65 ± 0.20
	MLM-3DAE	② 1.62 ± 0.05	2.04 ± 0.07	② 5.23 ± 0.22	1.09 ± 0.15	1.23 ± 0.21	① 3.53 ± 0.20
RMSE_{P}	MLM	N/A	N/A	33.40 ± 0.00	N/A	N/A	19.60 ± 0.00
	MLMp	N/A	N/A	69.63 ± 0.00	N/A	N/A	12.72 ± 0.00
	MLM-1DAE	N/A	N/A	② 15.10 ± 0.18	N/A	N/A	② 7.02 ± 0.26
	MLM-3DAE	N/A	N/A	① 14.87 ± 0.33	N/A	N/A	① 6.93 ± 0.29

TABLE II

Comparison of unmixing results $\times 10^{-2}$ (mean ± standard deviation) on synthetic data with R=8, SNR=20dB and 30dB, generated by LMM, PPNMM and MLM models, averaged over 10 runs

Mix. / Unmix.		SNR=20 dB			SNR=30 dB		
		LMM	PPNMM	MLM	LMM	PPNMM	MLM
SAD_{end}	VCA	2.62 ± 0.00	2.02 ± 0.00	5.53 ± 0.00	① 0.75 ± 0.00	1.00 ± 0.00	3.30 ± 0.00
	MLMp	30.10 ± 0.00	29.91 ± 0.00	25.37 ± 0.00	6.08 ± 0.00	6.55 ± 0.00	3.03 ± 0.00
	EndNet	5.43 ± 0.01	4.23 ± 0.01	9.91 ± 0.01	3.62 ± 0.02	3.51 ± 0.03	6.83 ± 0.02
	CNNAEU	2.49 ± 0.01	1.94 ± 0.01	5.31 ± 0.01	0.89 ± 0.01	1.23 ± 0.03	3.21 ± 0.01
	3D-NAE	2.51 ± 0.02	1.95 ± 0.04	① 4.73 ± 0.02	0.79 ± 0.03	① 0.97 ± 0.02	3.03 ± 0.03
	PPNMM-AE	2.55 ± 0.00	1.96 ± 0.00	5.23 ± 0.00	0.90 ± 0.00	1.14 ± 0.00	3.12 ± 0.00
	MLM-1DAE	① 2.33 ± 0.03	② 1.75 ± 0.04	② 4.99 ± 0.01	② 0.77 ± 0.01	① 0.97 ± 0.02	② 2.93 ± 0.01
	MLM-3DAE	② 2.35 ± 0.02	① 1.73 ± 0.04	5.00 ± 0.02	② 0.77 ± 0.01	0.98 ± 0.03	① 2.92 ± 0.01
$\text{RMSE}_{\text{abun}}$	FCLS	4.22 ± 0.00	3.80 ± 0.00	11.02 ± 0.00	1.92 ± 0.00	1.99 ± 0.00	10.24 ± 0.00
	MLM	4.94 ± 0.00	4.56 ± 0.00	5.86 ± 0.00	2.26 ± 0.00	2.31 ± 0.00	① 4.48 ± 0.00
	MLMp	4.59 ± 0.00	4.26 ± 0.00	9.51 ± 0.00	2.03 ± 0.00	2.02 ± 0.00	7.66 ± 0.00
	PPNMM	4.65 ± 0.00	4.35 ± 0.00	9.51 ± 0.00	2.33 ± 0.00	2.34 ± 0.00	8.25 ± 0.00
	EndNet	11.97 ± 0.03	12.07 ± 0.02	12.55 ± 0.04	11.97 ± 0.04	12.06 ± 0.01	12.48 ± 0.01
	CNNAEU	5.29 ± 0.17	5.25 ± 0.27	7.72 ± 1.50	4.41 ± 0.19	3.25 ± 0.44	6.68 ± 0.45
	3D-NAE	5.04 ± 0.07	4.84 ± 0.74	11.40 ± 0.17	4.60 ± 0.70	4.36 ± 0.67	11.12 ± 0.15
	PPNMM-AE	4.58 ± 0.00	3.87 ± 0.00	12.37 ± 0.00	6.90 ± 0.00	6.46 ± 0.00	11.92 ± 0.00
	MLM-1DAE	② 3.40 ± 0.15	② 2.72 ± 0.13	② 5.27 ± 0.22	② 1.80 ± 0.05	② 1.77 ± 0.09	5.06 ± 0.24
	MLM-3DAE	① 3.40 ± 0.08	① 2.67 ± 0.12	① 4.68 ± 0.19	① 1.77 ± 0.07	① 1.76 ± 0.08	② 4.69 ± 0.31
RMSE_{P}	MLM	N/A	N/A	23.95 ± 0.00	N/A	N/A	21.69 ± 0.00
	MLMp	N/A	N/A	69.33 ± 0.00	N/A	N/A	12.66 ± 0.00
	MLM-1DAE	N/A	N/A	② 6.75 ± 0.47	N/A	N/A	② 6.91 ± 0.38
	MLM-3DAE	N/A	N/A	① 5.30 ± 0.37	N/A	N/A	① 5.81 ± 0.25



(a) End#1 (b) End#2 (c) End#3 (d) End#4

Fig. 4. Estimated endmembers by different unmixing methods (the best in ten runs) on MLM-generated data at SNR=30 and R=4.

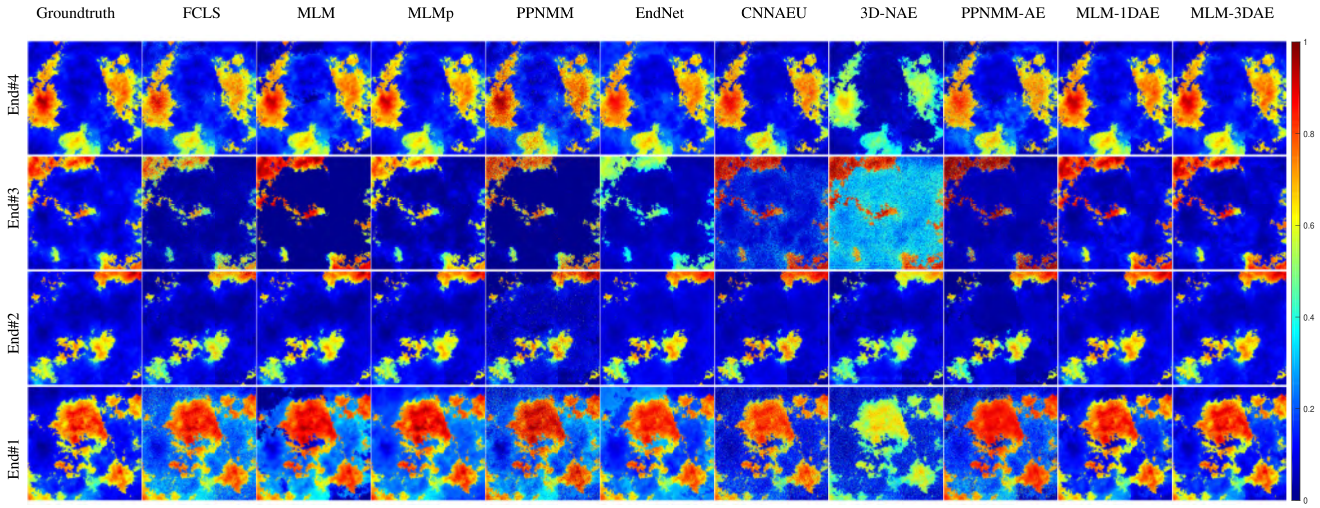
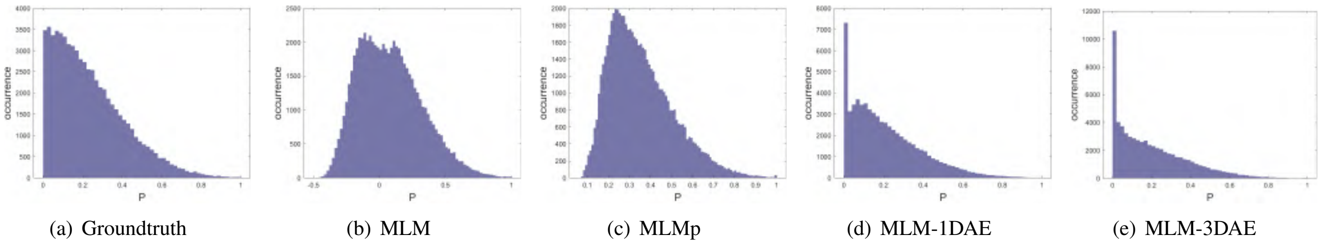


Fig. 5. Estimated abundance maps on MLM-generated data at SNR=30 and R=4. From left to right: Groundtruth, FCLS, MLM, MLMp, PPNMM, EndNet, CNNAEU, 3D-NAE, PPNMM-AE, MLM-1DAE, and MLM-3DAE.



(a) Groundtruth (b) MLM (c) MLMp (d) MLM-1DAE (e) MLM-3DAE

Fig. 6. Histograms of the transition probability P value obtained using MLM-derived methods, on MLM-generated data at SNR=30 and R=4.

Regarding the estimated abundance RMSEs achieved by different methods, the proposed techniques demonstrate strong performance, especially in scenarios with R=8 and on datasets generated by MLM. They consistently rank among the top two performers across different noise levels and endmember counts. Furthermore, it is noteworthy that, in most cases, MLM-3DAE outperforms MLM-1DAE in abundance estimation. This is attributed to the effective utilization of spectral-spatial dependencies present in the hyperspectral imagery in MLM-3DAE. Fig. 5 and Fig. 7 provide the abundance maps from various methods, specifically on MLM-generated datasets with SNR=30, R=4, and SNR=20, R=8, respectively. As observed, the MLM-derived unmixing methods generally yield abundance maps that visually align closely with the groundtruth. In the case with SNR=20 and R=8, the proposed methods exhibit enhanced robustness, yielding visually consistent abundance maps, while the abundance maps generated by the MLMp counterpart notably deteriorate.

Concerning the estimation of transition probability P using MLM-derived methods, the proposed MLM-1DAE and MLM-3DAE consistently outperform both MLM and MLMp across different endmember counts and varying noise levels. Notably, when compared to the primary counterpart MLMp, the proposed methods demonstrate better robustness in estimating P. This is particularly evident given that MLMp yields significantly deteriorated P-estimation results when SNR=20. Fig. 6 depicts the histogram of the distribution of P using different MLM-based methods, on data generated by MLM at SNR=30 and R=4. As observed, MLM-1DAE and MLM-3DAE offer the closest approximations to the true distribution of P, visually resembling a skewed half-normal distribution closely aligned with the ground truth.

B. *Experiments with Real Data*

1) Samson [20], [30]: The first real hyperspectral image is the widely-used Samson data, with a size of 95×95 pixels. After removing the noisy bands, 156 spectral bands are retained for analysis, covering the wavelength range from 401 nm to 889 nm. This dataset consists of three distinct endmembers, namely Soil, Water, and Tree.

Due to the smaller image scale, a batch size of 64 is employed for this dataset. The learning rate for the linear decoder within the Adam optimizer is set to 10^{-6}, while the remaining components of the network maintain a constant learning rate of 10^{-4}. The epoch number is set to 200.

Fig. 9 visualizes the endmember curves extracted by using different blind unmixing techniques, as well as the groundtruth endmembers provided in [51]. Fig. 10 depicts the abundance maps obtained using various unmixing methods on the Samson dataset, effectively highlighting the separation of three dominant materials achieved by these methods. Fig. 12 visualizes the histograms of the estimated transition probability P from different MLM-derived methods. As observed, both MLMp and the proposed methods yield very similar P-maps. In areas

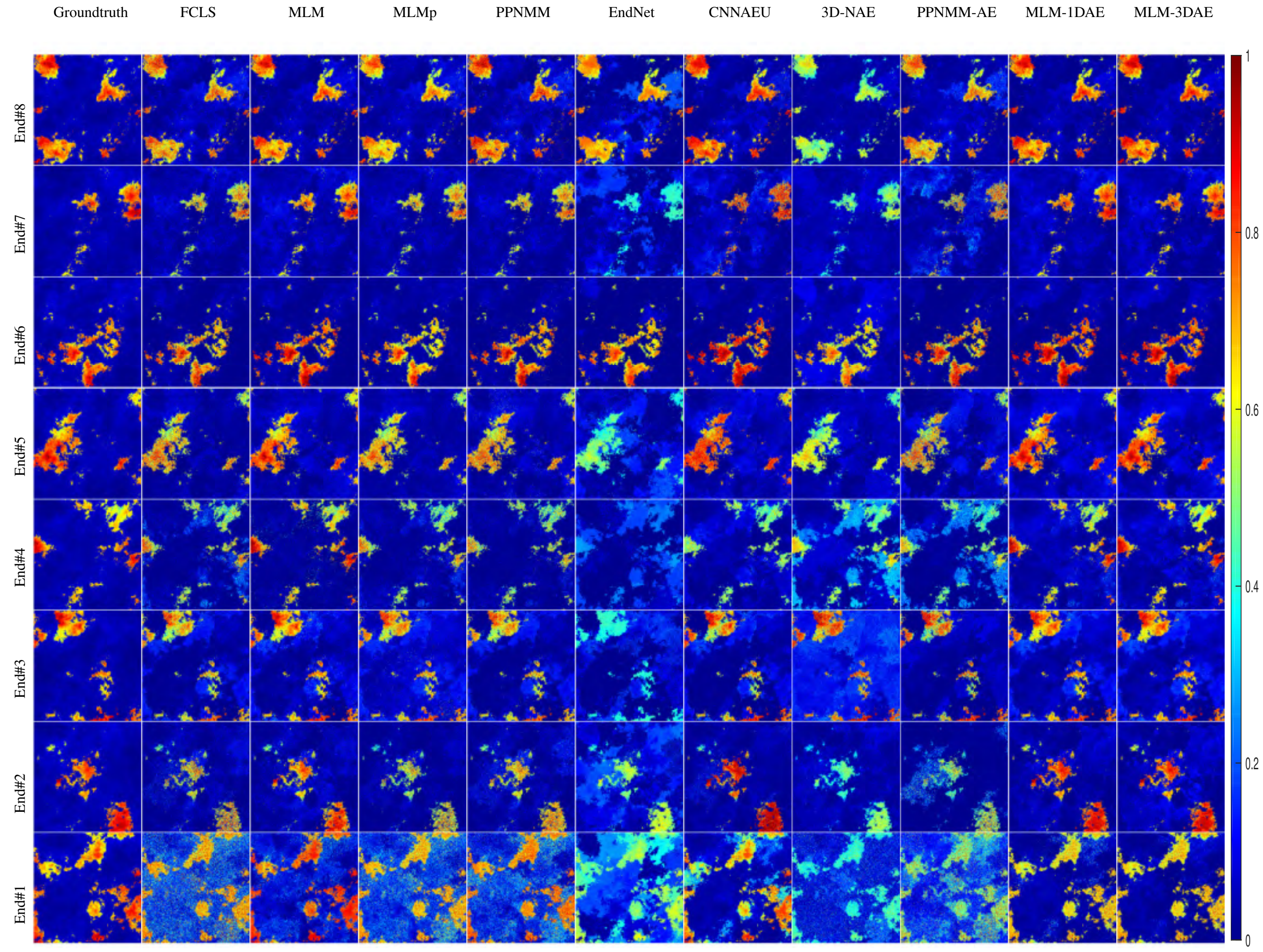


Fig. 7. Estimated abundance maps on MLM-generated data at SNR=20 and R=8. From left to right: Groundtruth, FCLS, MLM, MLMp, PPNMM, EndNet, CNNAEU, 3D-NAE, PPNMM-AE, MLM-1DAE, and MLM-3DAE.

TABLE III
COMPARISON OF PIXEL SAD $\times 10^{-2}$ (MEAN $\pm$ STANDARD DEVIATION) ON SAMSON, AVERAGED OVER 10 RUNS

FCLS	MLM	MLMp	PPNMM	EndNet	CNNAEU	3D-NAE	PPNMM-AE	MLM-1DAE	MLM-3DAE
5.97±0.00	4.42±0.00	4.91±0.00	8.51±0.00	5.13±0.32	11.23±1.57	8.46±0.44	5.36±0.00	②4.41±0.34	①4.18±0.30

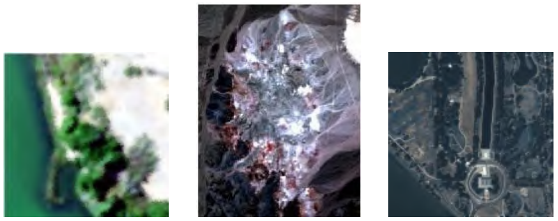

(a) Samson (b) Cuprite (c) Washington DC

Fig. 8. Three real hyperspectral images used in the experiments.

where water and soil intersect and at the boundaries between soil and trees, the estimated P values tend to be relatively larger, suggesting a higher degree of nonlinearity in these specific regions.

As the underlying mixing mechanism and groundtruth abundance are unavailable, we report the pixel SAD values only for reference in TABLE III, where the proposed methods achieve the lowest two values among the algorithms examined. Fig. 11 visualizes the spatial pixel RMSE maps by various algorithms, where the pixel RMSE is similarly defined as in (12).

2) *Cuprite [30]:* The second real data is the Cuprite data, which was taken over the Cuprite NV, USA, captured by the AVIRIS. The original image consists of 224 spectral bands, ranging in wavelengths from 370 nm to 2480 nm, with a spatial resolution of 20 meters. A sub-image of 250 × 190 pixels is selected from the original data, and 187 relatively clean bands are remained for analysis, after removing the ones influenced by water vapor absorption. A total R=12 prominent materials are considered for this dataset.

On this dataset, the batch size is set to 512, and the number of epochs is set to 300. The learning rate of linear decoder in

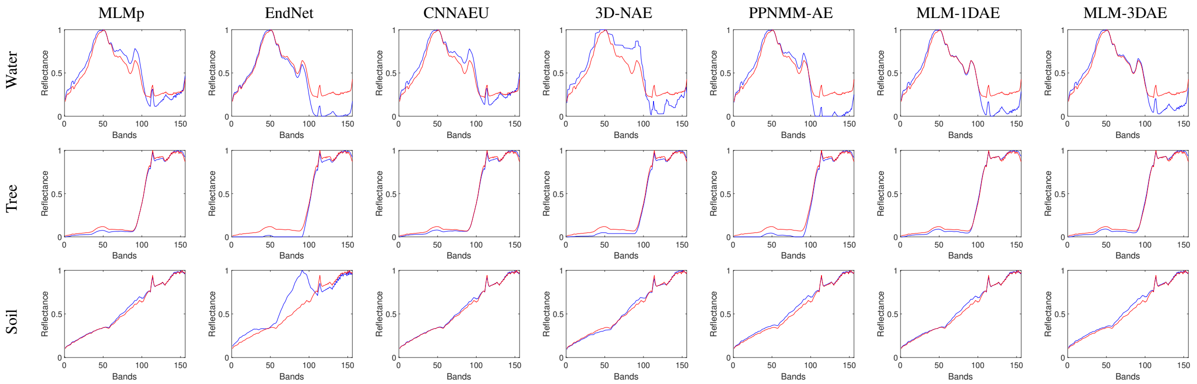


Fig. 9. Estimated endmembers by different unsupervised unmixing methods and the corresponding groundtruth (in red) on Samson data. Up to bottom: Water, Tree, and Soil. Left to right: MLMp, EndNet, CNNAEU, 3D-NAE, PPNMM-AE, MLM-1DAE, and MLM-3DAE.

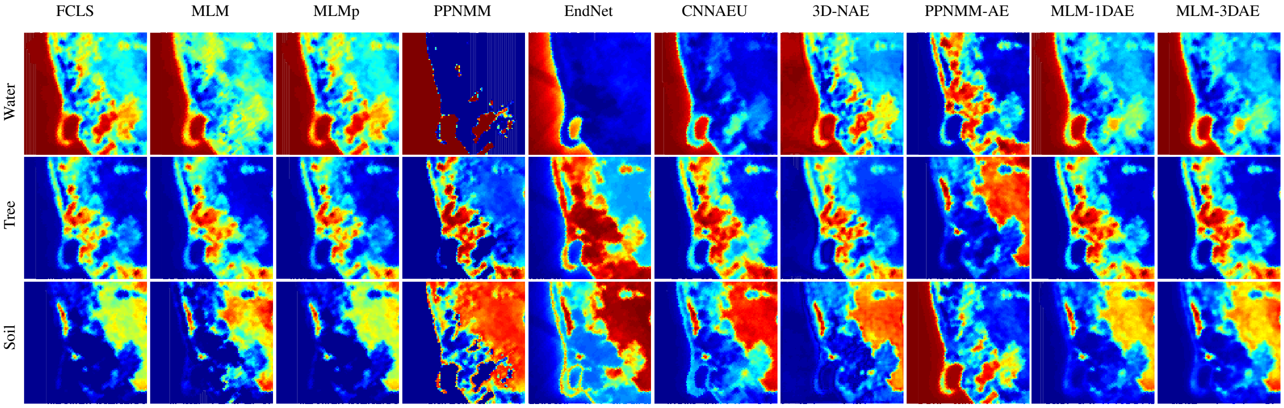


Fig. 10. Estimated abundance maps for different endmembers on Samson data. Up to bottom: Water, Tree, and Soil. Left to right: FCLS, MLM, MLMp, PPNMM, EndNet, CNNAEU, 3D-NAE, PPNMM-AE, MLM-1DAE, and MLM-3DAE.

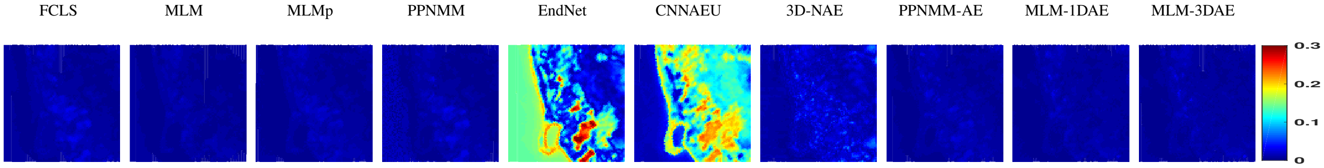
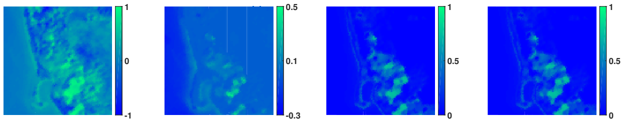


Fig. 11. Estimated pixel RMSE maps on Samson data. Left to right: FCLS, MLM, MLMp, PPNMM, EndNet, CNNAEU, 3D-NAE, PPNMM-AE, MLM-1DAE, and MLM-3DAE.

TABLE IV
COMPARISON OF PIXEL SAD $\times 10^{-2}$ (MEAN $\pm$ STANDARD DEVIATION) ON CUPRITE, AVERAGED OVER 10 RUNS

FCLS	MLM	MLMp	PPNMM	EndNet	CNNAEU	3D-NAE	PPNMM-AE	MLM-1DAE	MLM-3DAE
1.96±0.00	1.56±0.00	1.60±0.00	1.75±0.00	①1.22±0.06	13.60±2.01	1.96±0.06	8.21±0.00	②1.46±0.12	1.54±0.13



(a) MLM (b) MLMp (c) MLM-1DAE (d) MLM-3DAE

Fig. 12. Visualization of transition probability P value by using MLM, MLMp, MLM-1DAE and MLM-3DAE on Samson data.

the Adam optimizer is set to 10^{-5} . Meanwhile, we maintained a constant learning rate of 10^{-3} for the rest of the network.

In Fig. 13, the extracted endmembers are compared with the groundtruh endmembers [51] using various algorithms on the Cuprite dataset. Fig. 14 visualizes the estimated abundance maps obtained by different methods on Cuprite, where the proposed MLM-1DAE and MLM-3DAE provide clear abundance maps visually similar to other competitors. Fig. 16 visualizes P-maps, indicating the presence of nonlinear mixing in complex regions of the Cuprite dataset. TABLE IV reports the pixel

TABLE V

COMPARISON OF PIXEL SAD $\times 10^{-2}$ (MEAN ± STANDARD DEVIATION) ON WASHINGTON DC, AVERAGED OVER 10 RUNS

FCLS	MLM	MLMp	PPNMM	EndNet	CNNAEU	3D-NAE	PPNMM-AE	MLM-1DAE	MLM-3DAE
1.83 ± 0.00	1.61 ± 0.00	1.61 ± 0.00	①$1.53\pm0.00$	1.55 ± 0.14	2.55 ± 0.07	1.60 ± 0.03	①$1.50\pm0.00$	1.88 ± 0.14	1.88 ± 0.12

SAD by different methods for reference purpose. The proposed MLM-1DAE results in the second lowest SAD, inferior to EndNet. Fig. 15 visualizes the spatial pixel RMSE maps for different methods, offering insight into the performance of different methods across the spatial domain.

3) Washington DC [23]: The third real data is Washington DC, which was acquired by the HYDICE sensor. The original image comprises 210 spectral bands, spanning a wavelength range from 400 nm to 2400 nm. For this study, a subimage of size 256×256 pixels was chosen from the top-left corner. After eliminating the bands affected by water vapor and noise, 191 bands remain. The five identified endmembers are Grass, Water, Roof, Road, and Tree.

On this dataset, the batch size is set to 64, with a total of 300 epochs. The learning rate for the linear decoder is 10^{-7}, while a learning rate of 10^{-4} is used for the remaining parts.

In Fig. 17, we compare the extracted endmember curves obtained through various unmixing techniques with the groundtruth signatures provided in CyCU-Net [23]. The authors of CyCU-Net proposed a groundtruth estimation for the Washington DC dataset. In our analysis, we solely utilized the endmember groundtruth, choosing not to employ the abundance groundtruth from the paper, given its reliance on a linear mixing assumption. Fig. 18 illustrates the estimated abundance maps produced by different methods, showing that MLM-1DAE and MLM-3DAE yield abundance maps visually similar to other competitors. However, it is crucial to note that the resulting abundance maps for different materials exhibit challenges in full separation on this complex and large-scale dataset. Materials with similar endmembers, such as Grass and Tree, show less sparsity in abundance representations. This limitation may be attributed to the optimization of the nonlinear decoder of the proposed nonlinear unmixing network, influencing abundance sparsity. It could be mitigated by incorporating sparsity-promoting regularization terms into the current MLM-1DAE and MLM-3D networks. Fig. 19 illustrates P-maps, providing a relative nonlinear intensity distribution within the image. TABLE V reports the pixel SAD by different methods for reference purpose, where the two PPNMM-based unmixing techniques achieve the best results in terms of pixel SAD on this dataset.

C. Computational Complexity

We compared the memory consumption and running time of MLM-1DAE and MLM-3DAE to four existing unmixing networks: EndNet, CNNAEU, 3D-NAE, and PPNMM-AE. All experiments were conducted on an NVIDIA GeForce RTX 2080 Ti GPU. Our methods, CNNAEU, and 3D-NAE were implemented in PyTorch, while EndNet and PPNMM-AE were implemented using TensorFlow. TABLE VI and TABLE VII present the parameter scale in megabytes (MB) and running time, respectively, for each unmixing network applied to three real images. Despite the inherent complexity of nonlinear-based methods in comparison to linear-based ones (such as EndNet and CNNAEU), our proposed methods exhibit efficient memory usage with moderate parameter scales but extended running times when contrasted with linear-based unmixing networks. In particular, when compared to MLM-3DAE, MLM-1DAE demonstrates low sensitivity to increased dataset size. It is crucial to note that the running time is influenced by various factors, including data size, learning rate, batch size, and the deep learning framework employed.

D. MLM-1DAE v.s. MLM-3DAE

Regarding the network structure, the primary difference between the two modes lies in the encoders, as explicitly discussed in Section III-B. While MLM-1DAE utilizes 1D convolution to leverage spectral information, MLM-3DAE explores spectral-spatial correlations within data patches by 3D convolution. This variance in encoder implies the incorporation of different prior information, yielding distinct regularization effects on the estimated abundances. However, the price of 3D mode is an increased network scale with extended runtime, as reported in TABLE VI and TABLE VII, respectively.

As for unmixing performance, both modes exhibit similar endmember extraction results on synthetic datasets. Nevertheless, when it comes to abundance estimation and determining the transition probability P, MLM-3DAE outperforms MLM-1DAE in 10 out of the total 12 scenarios for the former and in all the MLM-generated scenarios for the latter. This demonstrates the effectiveness of MLM-3DAE in exploring spatial-spectral information through 3D convolution.

V. CONCLUSION

This paper proposed an autoencoder-based unmixing method for MLM, where endmembers, abundance vector, and transition probability parameter were estimated simultaneously. Taking advantage of elaborate network design, the unmixing process of MLM was realized explicitly. The proposed unmixing network had an encoder-decoder architecture: the encoder compressed the input pixel (image patch) into the low-dimensional abundance representation; the decoder successively imitated the linear part and built upon it, the MLM mechanism. The transition probability parameter was obtained by using a softmax operation on a high-level feature. We considered two modes in the proposed method: MLM-1DAE accounting for only spectral information, and MLM-3DAE incorporating the spectral-spatial information among the neighboring pixels. The effectiveness of the proposed method was proved on both the synthetic and real datasets,

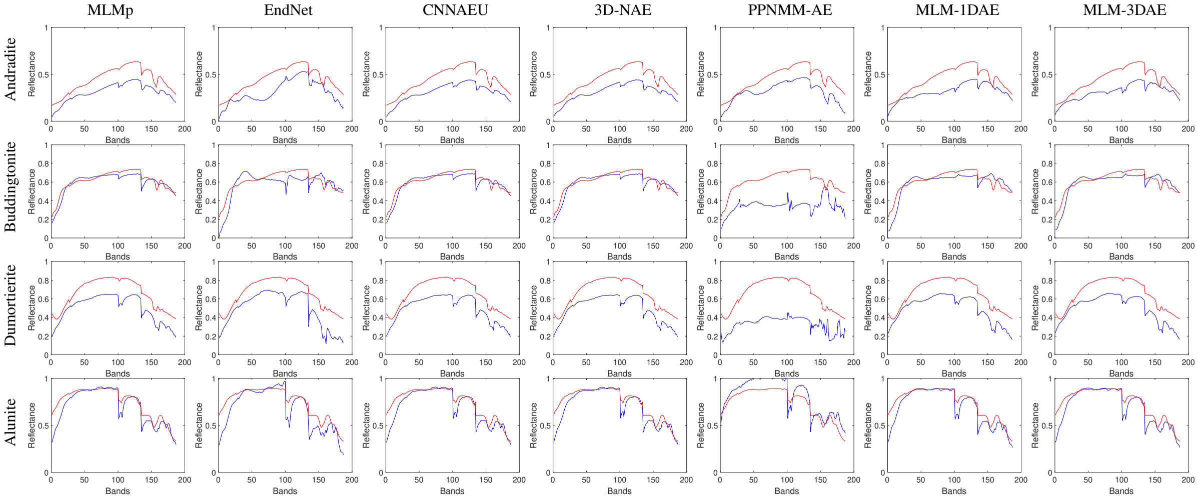


Fig. 13. Estimated 4 (out of a total 12) endmembers by different unsupervised unmixing methods and the corresponding groundtruth (in red) on Cuprite data. Up to bottom: Andradite, Buddingtonite, Dumortierite, and Alunite. Left to right: MLMp, EndNet, CNNAEU, 3D-NAE, PPNMM-AE, MLM-1DAE, and MLM-3DAE.

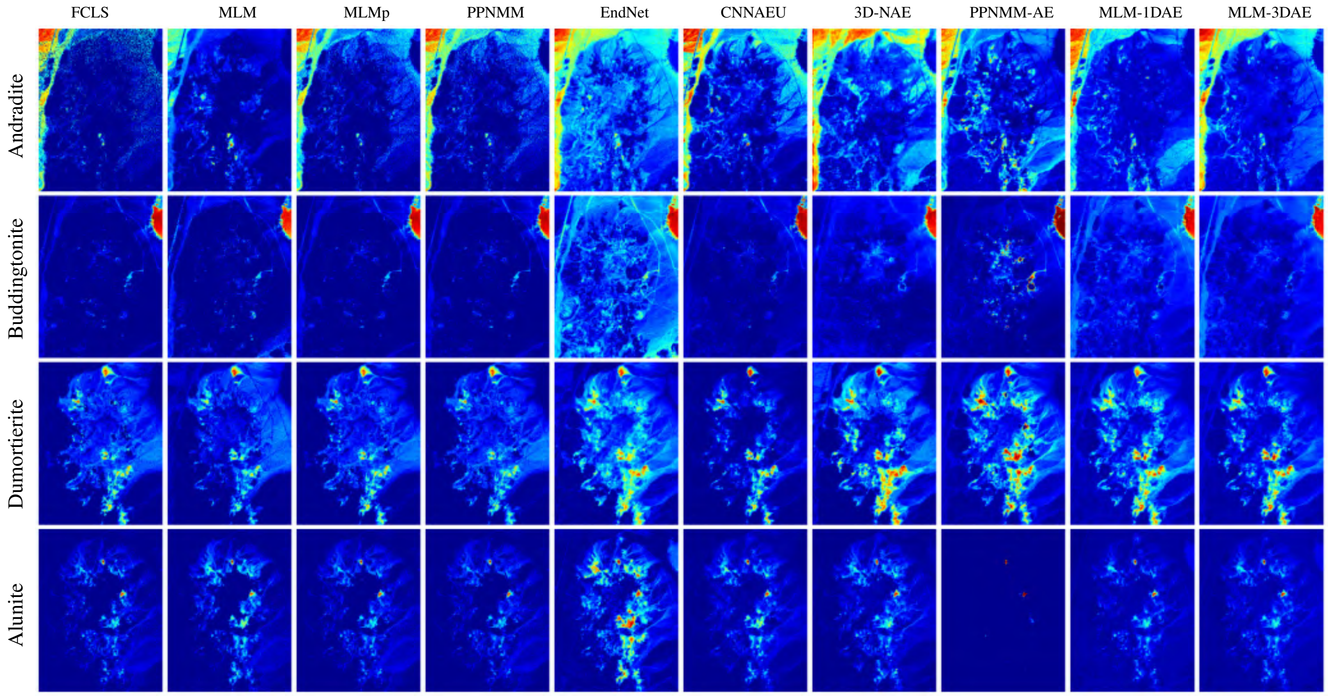


Fig. 14. Estimated abundance maps for 4 (out of a total 12) endmembers on Cuprite data. Up to bottom: Andradite, Buddingtonite, Dumortierite, Alunite. Left to right: FCLS, MLM, MLMp, PPNMM, EndNet, CNNAEU, 3D-NAE, PPNMM-AE, MLM-1DAE, and MLM-3DAE.

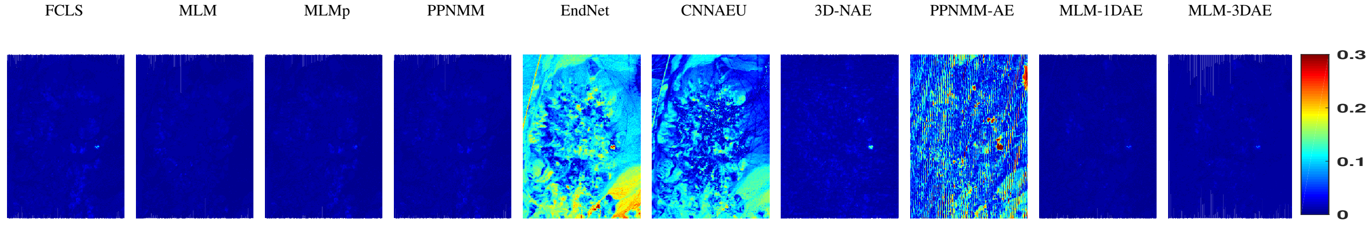


Fig. 15. Estimated pixel RMSE maps on Cuprite data. Left to right: FCLS, MLM, MLMp, PPNMM, EndNet, CNNAEU, 3D-NAE, PPNMM-AE, MLM-1DAE, and MLM-3DAE.

TABLE VI

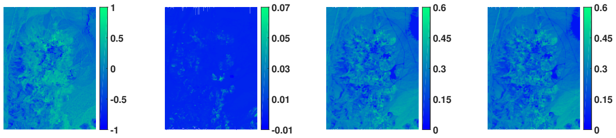
PARAMETER SCALE IN MEGABYTES (MB) OF THE COMPARING UNMIXING NETWORKS ON SAMSON, CUPRITE, AND WASHINGTON DC DATASETS

	EndNet	CNNAEU	3D-NAE	PPNMM-AE	MLM-1DAE	MLM-3DAE
Samson	0.0036	0.7320	2.8316	0.0380	0.2515	0.2753
Cuprite	0.0173	1.6742	4.2060	0.1078	0.4170	0.7759
Washington DC	0.0073	1.0726	3.3597	0.0386	0.3833	0.4473

TABLE VII

RUNNING TIME IN SECONDS (S) OF THE COMPARING UNMIXING NETWORKS ON SAMSON, CUPRITE, AND WASHINGTON DC DATASETS

	EndNet	CNNAEU	3D-NAE	PPNMM-AE	MLM-1DAE	MLM-3DAE
Samson	≈ 55	≈ 70	≈ 301	≈ 81	≈ 317	≈ 324
Cuprite	≈ 416	≈ 103	≈ 3282	≈ 371	≈ 748	≈ 1540
Washington DC	≈ 621	≈ 505	≈ 2717	≈ 223	≈ 1018	≈ 2003



(a) MLM (b) MLMp (c) MLM-1DAE (d) MLM-3DAE

Fig. 16. Visualization of transition probability P value by using MLM, MLMp, MLM-1DAE, and MLM-3DAE on Cuprite data.

through a comparative study with the MLM-based solutions and other unmixing networks.

In the future, we will extend our transition probability estimation to cover cases with $P < 0$ and investigate efficient convolution configurations, including separate 1D and 2D convolutions, to enhance the computational efficiency of MLM-3DAE in large-scale hyperspectral unmixing tasks.

REFERENCES

[1] N. Dobigeon, J.-Y. Tourneret, C. Richard, J. C. M. Bermudez, S. McLaughlin, and A. O. Hero, "Nonlinear unmixing of hyperspectral images: Models and algorithms," *IEEE Signal Proc. Mag.*, vol. 31, no. 1, pp. 82–94, 2014.

[2] R. Heylen, M. Parente, and P. Gader, "A review of nonlinear hyperspectral unmixing methods," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 7, no. 6, pp. 1844–1868, 2014.

[3] J. M. Bioucas-Dias, A. Plaza, N. Dobigeon, M. Parente, Q. Du, P. Gader, and J. Chanussot, "Hyperspectral unmixing overview: Geometrical, statistical, and sparse regression-based approaches," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 5, no. 2, pp. 354–379, 2012.

[4] B. Hapke, "Bidirectional reflectance spectroscopy: 1. Theory," *Journal of Geophysical Research: Solid Earth*, vol. 86, no. B4, pp. 3039–3054, 1981.

[5] W. Fan, B. Hu, J. Miller, and M. Li, "Comparative study between a new nonlinear model and common linear model for analysing laboratory simulated-forest hyperspectral data," *Int. J. Remote Sens.*, vol. 30, no. 11, pp. 2951–2962, 2009.

[6] A. Halimi, Y. Altmann, N. Dobigeon, and J.-Y. Tourneret, "Unmixing hyperspectral images using the generalized bilinear model," in *Proc. IEEE Int. Geosci. Remote Sens. Symp. (IGARSS)*, 2011, pp. 1886–1889.

[7] Y. Altmann, A. Halimi, N. Dobigeon, and J.-Y. Tourneret, "Supervised nonlinear spectral unmixing using a postnonlinear mixing model for hyperspectral imagery," *IEEE Trans. Image Process.*, vol. 21, no. 6, pp. 3017–3025, 2012.

[8] 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, 2016.

[9] Q. Wei, M. Chen, J.-Y. Tourneret, and S. Godsill, "Unsupervised nonlinear spectral unmixing based on a multilinear mixing model," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 8, pp. 4534–4544, 2017.

[10] B. Yang and B. Wang, "Band-wise nonlinear unmixing for hyperspectral imagery using an extended multilinear mixing model," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 11, 2018.

[11] M. Li, F. Zhu, A. J. X. Guo, and J. Chen, "A graph regularized multilinear mixing model for nonlinear hyperspectral unmixing," *Remote Sens.*, vol. 11, no. 19, 2019.

[12] B. Yang, "Supervised nonlinear hyperspectral unmixing with automatic shadow compensation using multiswarm particle swarm optimization," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, pp. 1–18, 2022.

[13] L. Gao, Z. Wang, L. Zhuang, H. Yu, B. Zhang, and J. Chanussot, "Using low-rank representation of abundance maps and nonnegative tensor factorization for hyperspectral nonlinear unmixing," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, pp. 1–17, 2022.

[14] Y. Gu, S. Wang, and X. Jia, "Spectral unmixing in multiple-kernel hilbert space for hyperspectral imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 51, no. 7, pp. 3968–3981, 2013.

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

[16] F. Zhu and P. Honeine, "Biobjective nonnegative matrix factorization: Linear versus kernel-based models," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 7, pp. 4012–4022, 2016.

[17] M. Li, B. Yang, and B. Wang, "Spectral–spatial reweighted robust nonlinear unmixing for hyperspectral images based on an extended multilinear mixing model," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, pp. 1–17, 2022.

[18] 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 Proc. Mag.*, vol. 40, no. 2, pp. 61–74, 2023.

[19] D. Bank, N. Koenigstein, and R. Giryes, "Autoencoders," *Machine Learning for Data Science Handbook: Data Mining and Knowledge Discovery Handbook*, pp. 353–374, 2023.

[20] 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, 2021.

[21] B. Palsson, J. Sigurdsson, J. R. Sveinsson, and M. O. Ulfarsson, "Hyperspectral unmixing using a neural network autoencoder," *IEEE Access*, vol. 6, pp. 25 646–25 656, 2018.

[22] Z. Han, D. Hong, L. Gao, B. Zhang, M. Huang, and J. Chanussot, "Autonas: Automatic neural architecture search for hyperspectral unmixing," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, pp. 1–14, 2022.

[23] 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, pp. 1–14, 2022.

[24] Y. Su, A. Marinoni, J. Li, J. Plaza, and P. Gamba, "Stacked nonnegative sparse autoencoders for robust hyperspectral unmixing," *IEEE Geosci. Remote Sens. Lett.*, vol. 15, no. 9, pp. 1427–1431, 2018.

[25] Y. Qu, R. Guo, and H. Qi, "Spectral unmixing through part-based nonnegative constraint denoising autoencoder," in *Proc. IEEE Int. Geosci. Remote Sens. Symp. (IGARSS)*, 2017, pp. 209–212.

[26] 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, 2018.

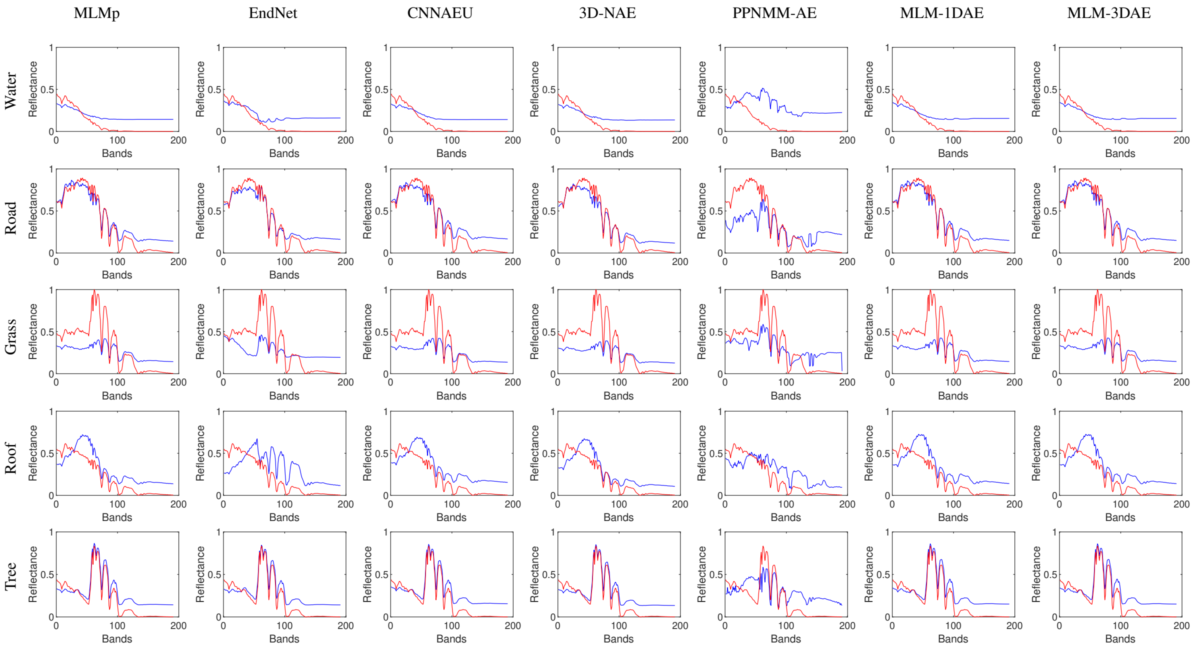


Fig. 17. Estimated endmembers by different unsupervised unmixing methods and the corresponding groundtruth (in red) on Washington DC data. Up to bottom: Water, Roof, Grass, Road, and Tree. Left to right: MLMp, EndNet, CNNAEU, 3D-NAE, PPNMM-AE, MLM-1DAE, and MLM-3DAE.

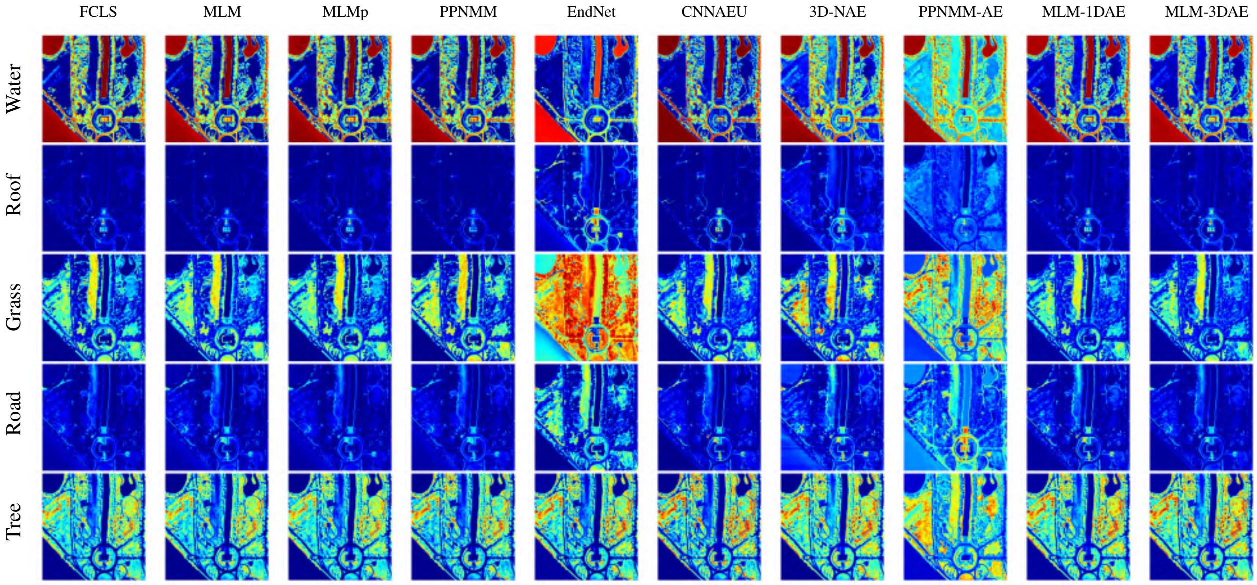
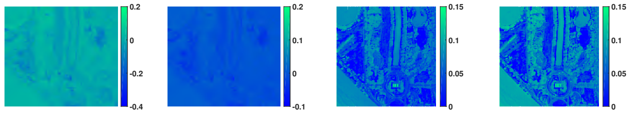


Fig. 18. Estimated abundance maps for different endmembers on Washington DC data. Up to bottom: Water, Roof, Grass, Road, Tree. Left to right: FCLS, MLM, MLMp, PPNMM, EndNet, CNNAEU, 3D-NAE, PPNMM-AE, MLM-1DAE, and MLM-3DAE.



(a) MLM (b) MLMp (c) MLM-1DAE (d) MLM-3DAE

Fig. 19. Visualization of transition probability P value by using MLM, MLMp, MLM-1DAE, and MLM-3DAE on Washington DC data.

[27] Q. Jin, Y. Ma, F. Fan, J. Huang, X. Mei, and J. Ma, "Adversarial autoencoder network for hyperspectral unmixing," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 34, no. 8, pp. 4555–4569, 2023.

[28] B. Rasti, B. Koirala, P. Scheunders, and J. Chanussot, "MiSiCNet: Minimum simplex convolutional network for deep hyperspectral unmixing," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, pp. 1–15, 2022.

[29] 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, 2019.

[30] 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.

[31] X. Xu, X. Song, T. Li, Z. Shi, and B. Pan, "Deep autoencoder for hyperspectral unmixing via global-local smoothing," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, pp. 1–16, 2022.

[32] D. D. Lee and H. S. Seung, "Algorithms for non-negative matrix factorization," in *Proc. Adv. Neural Inf. Process. Syst.*, 2001, pp. 556–562.

[33] X.-R. Feng, H.-C. Li, R. Wang, Q. Du, X. Jia, and A. Plaza, "Hyperspectral unmixing based on nonnegative matrix factorization: A comprehensive review," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 15, pp. 4414–4436, 2022.

[34] Y. Qian, F. Xiong, Q. Qian, and J. Zhou, "Spectral mixture model inspired network architectures for hyperspectral unmixing," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 10, pp. 7418–7434, 2020.

[35] C. Cui, X. Wang, S. Wang, L. Zhang, and Y. Zhong, "Unrolling nonnegative matrix factorization with group sparsity for blind hyperspectral unmixing," *IEEE Trans. Geosci. Remote Sens.*, vol. 61, pp. 1–12, 2023.

[36] F. Xiong, J. Zhou, S. Tao, J. Lu, and Y. Qian, "Snmf-net: Learning a deep alternating neural network for hyperspectral unmixing," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, pp. 1–16, 2022.

[37] 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, 2019.

[38] 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, pp. 1–15, 2022.

[39] M. Zhao, L. Yan, and J. Chen, "LSTM-DNN based autoencoder network for nonlinear hyperspectral image unmixing," *IEEE J. Sel. Topics Signal Process.*, vol. 15, no. 2, pp. 295–309, 2021.

[40] Y. Su, X. Xu, J. Li, H. Qi, P. Gamba, and A. Plaza, "Deep autoencoders with multitask learning for bilinear hyperspectral unmixing," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 10, pp. 8615–8629, 2020.

[41] K. T. Shahid and I. D. Schizas, "Unsupervised hyperspectral unmixing via nonlinear autoencoders," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, pp. 1–13, 2022.

[42] J. S. Bhatt, M. V. Joshi, and S. Vijayashekhar, "A multitemporal linear spectral unmixing: An iterative approach accounting for abundance variations," in *Proc. 9th Workshop Hyperspectral Image Signal Process., Evol. Remote Sens. (WHISPERS)*, 2018, pp. 1–5.

[43] L. Drumetz, M. D. Mura, G. Tochon, and R. Fablet, "Learning endmember dynamics in multitemporal hyperspectral data using a state-space model formulation," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process. (ICASSP)*, 2020, pp. 2483–2487.

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

[45] C.-I. Chang, "A review of virtual dimensionality for hyperspectral imagery," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 11, no. 4, pp. 1285–1305, 2018.

[46] V. S S, J. S. Bhatt, and B. Chattopadhyay, "Virtual dimensionality of hyperspectral data: Use of multiple hypothesis testing for controlling type-i error," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 13, pp. 2974–2985, 2020.

[47] J. M. Bioucas-Dias and J. M. P. Nascimento, "Hyperspectral subspace identification," *IEEE Trans. Geosci. Remote Sens.*, vol. 46, no. 8, pp. 2435–2445, 2008.

[48] D. Hong, L. Gao, J. Yao, N. Yokoya, J. Chanussot, U. Heiden, and B. Zhang, "Endmember-guided unmixing network (egu-net): A general deep learning framework for self-supervised hyperspectral unmixing," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 33, no. 11, pp. 6518–6531, 2022.

[49] P. Ghosh, S. K. Roy, B. Koirala, B. Rasti, and P. Scheunders, "Hyperspectral unmixing using transformer network," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, pp. 1–16, 2022.

[50] D. Heinz and Chein-I-Chang, "Fully constrained least squares linear spectral mixture analysis method for material quantification in hyperspectral imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 39, no. 3, pp. 529–545, 2001.

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