Dynamical hyperspectral unmixing with variational recurrent neural networks

Ricardo Augusto Borsoi, Tales Imbiriba, Pau Closas

Dynamical Hyperspectral Unmixing with Variational Recurrent Neural Networks

Ricardo A. Borsoi, *Member, IEEE*, Tales Imbiriba, Pau Closas, *Senior Member, IEEE*

***Abstract*—Multitemporal hyperspectral unmixing (MTHU) is a fundamental tool in the analysis of hyperspectral image sequences. It reveals the dynamical evolution of the materials (endmembers) and of their proportions (abundances) in a given scene. However, adequately accounting for the spatial and temporal variability of the endmembers in MTHU is challenging, and has not been fully addressed so far in unsupervised frameworks. In this work, we propose an unsupervised MTHU algorithm based on variational recurrent neural networks. First, a stochastic model is proposed to represent both the dynamical evolution of the endmembers and their abundances, as well as the mixing process. Moreover, a new model based on a low-dimensional parametrization is used to represent spatial and temporal endmember variability, significantly reducing the amount of variables to be estimated. We propose to formulate MTHU as a Bayesian inference problem. However, the solution to this problem does not have an analytical solution due to the nonlinearity and non-Gaussianity of the model. Thus, we propose a solution based on deep variational inference, in which the posterior distribution of the estimated abundances and endmembers is represented by using a combination of recurrent neural networks and a physically motivated model. The parameters of the model are learned using stochastic backpropagation. Experimental results show that the proposed method outperforms state of the art MTHU algorithms.**

***Index Terms*—Hyperspectral data, hyperspectral unmixing, recurrent neural networks, deep learning, multitemporal.**

I. Introduction

Hyperspectral images (HIs) have very high spectral resolution, which allows for a precise discrimination of different materials in a scene [1]. However, physical limitations of spectral image acquisition and large distances between the sensor and the scene of interest as seen in, e.g., remote sensing, means that each pixel of an HI may cover a large area of the scene and typically contains a mixture of different materials [2]. Hyperspectral unmixing (HU) aims to decompose an HI into the spectral signatures of the pure materials it contains (the *endmembers* – EMs), and the proportions with which they appear in each pixel (the *abundances*) [3].

The classical approach to describe the interactions between light and the different materials in a pixel is the linear mixing model (LMM) [3]. However, the LMM assumes the EM signatures to be the same for all pixels in an HI, disregarding spectral variability of the EMs which can be caused by, e.g., atmospheric, illumination or seasonal variations, and propagates significant errors throughout the HU processing chain [4], [5]. Thus, significant effort has been dedicated to address spectral variability in HU (see Section II for a review).

More recently, multitemporal HU (MTHU) has been receiving increasing interest in the literature since it leverages information in sequences of HIs acquired at different time instants to reveal the dynamical evolution of the endmembers and abundances in a scene [6]–[9]. MTHU has proven important for many applications such as invasive species mapping in rainforests [10], [11], and monitoring vegetation cover in shrublands [12] or seasonal variations of vegetation cover in dry forests [13]. Moreover, MTHU is also useful to perform change detection at the subpixel level [14], [15]. However, spectral variability can be very significant in MTHU due to different seasonal and acquisition conditions [4], [5], [7].

Addressing both the spatial and temporal spectral variability of the EMs is challenging, and has only been done in MTHU by supervised techniques [10], [16]. However, supervised MTHU techniques require prior knowledge of libraries containing spectral signatures which can accurately represent the endmembers for each image in the time sequence. Such libraries can be difficult or expensive to collect. Unsupervised MTHU methods, on the other hand, estimate both the endmember signatures and the abundances for all time instants directly from the observed HI sequence. Thus, unsupervised methods are of great practical interest, but can be challenging to design. See Section II for a review of MTHU methods.

Machine learning has become a popular framework to solve the HU problem [17]. Recent developments include methods based on, e.g., autoencoders (AECs) [18] or unrolled optimization-based neural networks [19] (see Section II for a review). In particular, solutions based on deep learning are especially attractive for HU when the mixing model considers nonidealities such as, e.g., nonlinearity [20] and EM variability [21], circumventing the need to construct complex analytical models to represent such physical effects.

However, the literature lacks MTHU solutions that are unsupervised and take spatial and temporal EM variability into account, which are addressed in this work. In particular, we also address several other needs, including the development of parsimonious models for EM variability with adjustable flexibility, and of machine learning-based strategies for MTHU which jointly leverage both a physically motivated and data-driven (e.g., neural networks) models in a principled manner. Such hybrid approaches, where physics-informed models are used to regularize and provide interpretability to data-driven methodologies are becoming increasingly popular [22].

In this work, we propose an unsupervised MTHU algorithm based on variational recurrent neural networks (RNNs). First,

R.A. Borsoi is with the University of Lorraine, CNRS, CRAN, Nancy, F-54000, France (e-mail: {ricardo.borsoi}@univ-lorraine.fr).

T. Imbiriba and P. Closas are with the Dept. of Electrical & Computer Engineering, Northeastern University, Boston, MA 02115, USA (e-mail: {talesim,closas}@ece.neu.edu).

This work has been partially supported by the NSF under Award ECCS-1845833.

a stochastic model is proposed to represent both the dynamical evolution of the EMs and of the abundances, as well as the mixing process. Moreover, a new low-dimensional model is used to represent spatial and temporal EM variability by parametrizing band-dependent scaling variations of the EMs using a set of smooth spectral basis functions. This allows us to control the flexibility of the model by varying the number and types of basis functions. To model the abundances, we make use of the *softmax basis* representation [23], which leads to a physically accurate model and has been successfully used for fuzzy classification [24] and single-image HU [25]. This way, we can use a Gaussian distribution to represent the abundances in the softmax basis, which closely approximates a Dirichlet distribution when mapped back to the original abundance domain (i.e., the unit simplex) [26].

MTHU is then formulated as a Bayesian inference problem. However, exact inference is analytically intractable due to the nonlinearities in the model. Thus, we consider a variational inference solution based on RNNs, in which the approximate joint posterior distribution of abundances and EMs is learned by maximizing a lower bound over the marginal likelihood of all pixels. Note that approximating the true posterior typically requires a flexible family of distributions which can be represented using neural networks [27]. However, using feedforward neural networks leads to models with large numbers of parameters, making inference costly. By exploiting the temporal structure in the data (e.g., Markovity), RNNs provide a solution that gives flexibility while also having a lower number of parameters (being computationally lightweight). Besides, RNNs have shown excellent performance in numerous sequence modeling tasks [28]. Interpretability of the estimated abundances and EMs is paramount for the applicability of MTHU systems. For this reason, we parameterized the joint posterior distribution using a hybrid model composed of physics-based and data-driven components. More specifically, the posterior is modeled by a family of nonlinear functions constructed by integrating both a simple, physically motivated model that is able to provide an approximate abundance estimate, and a bidirectional RNN that can represent more complex effects (i.e. not captured by the simpler model). The parameters of the model and of the posterior distribution are learned based on all image pixels using stochastic gradient descent (SGD). The contributions of this paper include:

- a new low-dimensional model to represent the spatial and temporal variability of the EMs with a small amount of parameters to be learned;
- a stochastic model describing the temporal evolution of the abundances and of the EM variability parameters, which leverages the softmax basis used in single-image HU [25] to obtain a physically accurate representation of the abundance dynamics using a Gaussian distribution;
- a deep variational inference formulation of model-based MTHU with both spatial and temporal EM variability, solved using stochastic backpropagation;
- a parametrization of the posterior distribution of the abundances and EMs combining bidirectional RNNs and a physically interpretable model.

The proposed method is called ReDSUNN for *Recurrent hyperSpectral Unmixing with Neural Networks*. Experimental results with synthetic and real data show that ReDSUNN outperforms state of the art MTHU algorithms. Codes are available at https://github.com/ricardoborsoi/ReDSUNN.

II. Background and related work

A general multitemporal linear mixing model represents the n-th pixel of an HI acquired at time t as:

$$\begin{aligned} &\boldsymbol{y}_{n,t} = \boldsymbol{M}_{n,t}\boldsymbol{a}_{n,t} + \boldsymbol{r}_{n,t}\,, \\ &\text{s.t.}\;\; \mathbf{1}^\top \boldsymbol{a}_{n,t} = 1,\;\; \boldsymbol{a}_{n,t} \geqslant \mathbf{0}\,, \end{aligned} \tag{1}$$

where, for each time $t \in \{1,\dots,T\}$ and pixel $n \in \{1,\dots,N\}$, $\boldsymbol{y}_{n,t} \in \mathbb{R}^L$ denotes the observed pixel with L bands, the columns of $\boldsymbol{M}_{n,t} \in \mathbb{R}^{L\times P}$ contain the spectral signatures of the P endmembers in the scene, vector $\boldsymbol{a}_{n,t} \in \mathbb{R}^P$ contains the fractional abundances of each EM, and $\boldsymbol{r}_{n,t} \in \mathbb{R}^L$ represents additive noise. Note that the general model (1) can accommodate EM variability both in space and in time, being able to represent effects such as, e.g., atmospheric, illumination, or seasonal variations [4]. In the following, we review different HU strategies addressing multitemporal sequences, spatial EM variability, and based on deep learning frameworks.

A. Multitemporal HU

A fundamental aspect of MTHU is taking into account the relationship between the EMs and abundances at different time instants. Since these are usually temporally correlated, this can greatly improve the performance of unmixing algorithms. Most previous works have been focused on addressing the variability of the EMs in time. This was usually performed by considering a more constrained version of model (1), where only the temporal variability of the EMs is considered, leading to $\boldsymbol{M}_{n,t} = \boldsymbol{M}_{m,t}$, for all $1 \leqslant n, m \leqslant N$ [6]–[8].

Several works have considered parametric models to represent the temporal variability of the EMs (i.e., using only a single EM matrix per image). For instance, dynamical model was used in [6] to constrained the EMs to be a scaled versions of a reference EM matrix, with smoothly varying scaling factors. Another model considered the EM matrices at each time instant to be an additive perturbation of a mean EM matrix [7]. Using this model, MTHU was performed using a two-stage stochastic programming approach [7]. Other works considering this model have also proposed MTHU solutions using a distributed algorithm using sparsity constraints [29], and a hierarchical Bayesian framework incorporating additive residual terms to account for outliers [30]. A recent work proposed a hierarchical Bayesian MTHU strategy (called HBUN) which incorporated priors promoting the spectral smoothness of the estimated EM signatures, and the spatial and temporal smoothness of the abundances [9]. Another model representing the EMs using bandwise multiplicative scalings of a set of reference EM signatures was considered in [8]. MTHU was performed by combining a Bayesian filtering (the Kalman filter and smoother) with the expectation maximization method. Heteroscedastic measurement noise was also considered in [31], where the (diagonal) covariance matrix of

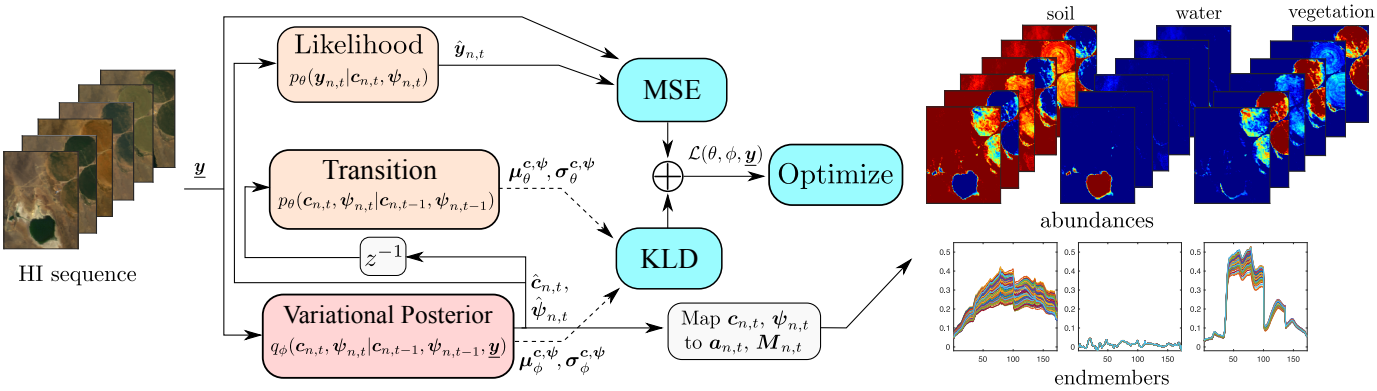

Fig. 1: Illustrative diagram of the proposed ReDSUNN method. A likelihood and transition PDFs, with parameters θ, define the spatiotemporal mixing process (i.e., the generative model). The variational posterior, with parameters ϕ, approximates the unmixing solution (i.e., the PDF of the abundances and EMs conditioned on the HIs) recursively, being implemented using an RNN. The parameters of these PDFs are learned by maximizing a loss function based on the ELBO, which balances data reconstruction (MSE) and consistency between posterior and prior (KLD). The parameters $\boldsymbol{c}_{n,t}$ and $\boldsymbol{\psi}_{n,t}$ are mapped to the abundances and EM matrices, $\boldsymbol{a}_{n,t}$ and $\boldsymbol{M}_{n,t}$ using a deterministic model. The notation z^{-1} represents a delay. See Section III for more details.

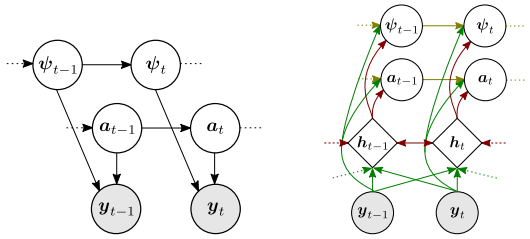

Fig. 2: In the generative model (left), conditional independence among pixels $\boldsymbol{y}_t$ is assumed given abundances $\boldsymbol{a}_t$ and variability coefficients $\boldsymbol{\psi}_t$, as well as Markovity of state variables. During inference (right), state estimation exploits correlations between pixels, states and hidden representations $\boldsymbol{h}_t$ generated by a bi-directional RNN.

the measurement noise was estimated along with the EMs and abundances in a *maximum a posteriori* framework.

However, these methods disregard EM variability within a single image, which limits their performance. Existing MTHU algorithms which tackle both spatial and temporal variabilty are based on the MESMA framework [32], which searches for a combination of EM signatures in a library which can best reconstruct a given HI, rendering it highly interpretable. These have been applied for MTHU in vegetation monitoring applications [10], [12], [33]. Recent advances were also made on developing efficient solutions to this problem with theoretical guarantees by exploring the temporal information of the abundances [16]. However, MESMA-based approaches are computationally costly and their performance depends strongly on the accuracy of the library, rendering this model inadequate for unsupervised processing. Note that some works proposed to leverage complementary high-resolution (Landsat) images to unmix low-resolution (MODIS) images [34], [35]. However, the availability of such complementary information and difficulties associated to differences in their acquisition times limit the applicability of such methods in practice.

B. HU with EM variability

The variability of the EMs across an HI occurs due to atmospheric, illumination (e.g., topographic) or intrinsic variations of the EM spectra [4], [5], [36]. It introduces errors which propagate through the various steps of the traditional HU processing chains and can have significant negative impact on the abundance estimation performance. EM variability is generally addressed in HU by representing the spectral signature of each material using structured spectral libraries, statistical distributions, or physically motivated parametric models [4].

Spectral library-based methods represent the EMs in each pixel as one of several spectral signatures in a dictionary known a priori, which implies formulating HU as a structured sparse regression problem. The sparsity prior can be addressed either with combinatorial approaches, which are computationally costly [32], or using different relaxations of the L_0 seminorm based on convex [37], [38] or non-convex [39] sparsity promoting penalties. Other strategies also allow each EM to be a convex combination of library spectra [40]. Such relaxations are computationally easier to solve. This can be a reasonable modeling assumption when multiple signatures of the same EM can contribute to form a single pixel. However, such relaxations might reduce the interpretability of the solutions if the goal is to identify which signature is active [39]. Nevertheless, the performance of both kinds of strategies is strongly dependent on the quality of the spectral library.

Using statistical distributions to model the endmembers has been well-investigated as it provides principled HU solutions through a Bayesian framework. The Gaussian [41], mixture of Gaussians [42] or Beta [43] distributions have been considered. However, when complex distributions (such as the Beta or mixtures of Gaussians) are used to represent the endmembers, HU (which consists in a Bayesian inference problem) can become computationally expensive.

Another approach to address spectral variability consists in representing the signatures of the EMs in each pixel using a physically meaningful parametric model, and estimating the model parameters during HU. Examples of such models include the use of additive perturbations [44], spectrally uniform or spectrally localized multiplicative scaling factors [45], [46], or a combination thereof [47]. Other models explicitly exploit spatial information, using multiscale [48] or low-rank tensor representation [49], and external information (e.g., LiDAR)

by means of digital surface models [50]. However, designing physically accurate models whose parameters can at the same time be properly recovered from an HI can be challenging.

C. Deep learning-based HU

Deep learning has recently become a popular approach to perform HU. While HU was traditionally viewed as a regression problem (i.e., learning a mapping from the pixels to the abundances) [51], [52], recent work has been focused on developing unsupervised or self-supervised strategies, which avoid the need for vast amounts of training data. Among such strategies, AECs have become a predominant approach for deep learning-based HU due to their close connection to linear or nonlinear mixing models and good experimental performance [18]. The latent representations of the image pixels obtained by the AEC are associated to the abundances, and the decoder network to the mixing model [53], [54]. Several AEC methods for HU have been proposed for linear HU by using different choices of encoder networks including, e.g., denoising layers [55], [56], spatial-spectral (convolutional) architectures [57], [58] and the use of sparsity constraints [59].

AECs have also been used to perform nonlinear HU by considering nonlinear decoder networks to address complex mixing effects. This includes a post-nonlinear mixing model [20], additive nonlinearities [60] and the use of application-specific nonlinear neural network layers [61]. Another work also exploited the relationship between the encoder and decoder networks to propose a model-based architecture [62].

Spectral variability was also addressed using deep learning methods. In [21], a generative EM model is proposed to represent the variability of the EMs on a low-dimensional manifold. Such a model was used to perform HU using methods inspired by matrix factorization [21], sparse regression [63] and probabilistic approaches [56], [64]. Gaussian process regression has also been considered as a non-parametric approach to mitigate the effects of spectral variability in HU [65], [66].

Other approaches also used multiple sets of pure pixels extracted in a self-supervised manner to regularize (explicitly or implicitly) AEC-based HU algorithms in order to improve their robustness [67]–[69]. An approach to learn learn dynamical models for the spectra of individual materials has also been proposed [70]. Finally, other methods have been proposed using, e.g., Wasserstein [71], adversarial [72], or cycle-consistency [73] loss functions during training. Deep priors [74] and unfolding optimization-based neural networks have also been considered [19].

We highlight, however, that despite these advances previous MTHU methods did not address spatial EM variability without supervision, and also did not exploit deep learning frameworks. In the following, we will present a probabilistic model representing both spatial and temporal EM variability, and develop an unsupervised inference strategy based on RNNs.

III. Overview of the proposed approach

We consider a probabilistic framework for MTHU. This amounts to two steps. The first is the *modeling step*, which consists in defining a set of probability density functions (PDFs) which describe how the abundances and EMs evolve over time, and how the pixels are generated. Note that the model PDFs typically depend on different deterministic parameters, some of which are specified a priori (which we refer to as *hyperparameters*) and some which we intend to learn from the observed HIs. We represent the parameters to be learned in the set θ. We include the subscript θ on the model PDFs in order to make their dependence on these parameters explicit.

The second is the *inference step*, which consists in computing (an approximation of) the posterior distribution, which is the PDF of the abundances and EMs conditioned on the observed pixels. The inference step is also decomposed in two distinct parts which are interdependent. The first part consists in computing the approximate posterior distribution, while the second part involves learning the deterministic parameters of the model in θ by maximizing the likelihood of the pixels.

Note that this process leads to an *unsupervised learning* problem, that is, the model parameters in θ and posterior distribution are both computed based only on the observed HI pixels (i.e., there is no separate training and testing data). In the following, we provide a high-level description of the approach, which is illustrated in Figures 1 and 2. Modeling and inference steps are then detailed in Sections IV and V.

Modeling step: The first step is to characterize the dynamical evolution of the EMs and abundances. Under a Markovity assumption, it can be expressed using the following sequence of conditional probability distributions [75]:

$$(\boldsymbol{a}_t, \boldsymbol{M}_t) \sim p_\theta(\boldsymbol{a}_t, \boldsymbol{M}_t | \boldsymbol{a}_{t-1}, \boldsymbol{M}_{t-1}) \,, \tag{2}$$

where $\boldsymbol{a}_t = [\boldsymbol{a}_{1,t}^\top, \ldots, \boldsymbol{a}_{N,t}^\top]^\top$ denotes the abundance maps in lexicographic ordering and $\boldsymbol{M}_t = \{\boldsymbol{M}_{1,1}, \ldots, \boldsymbol{M}_{N,t}\}$ the collection of EM matrices for all pixels, and $(\boldsymbol{a}_0, \boldsymbol{M}_0) \sim p_\theta(\boldsymbol{a}_0, \boldsymbol{M}_0)$. We also assume that the abundances and endmembers at time t are statistically independent when conditioned on the their values at time $t-1$, that is,

$$p_\theta(\boldsymbol{a}_t, \boldsymbol{M}_t | \boldsymbol{a}_{t-1}, \boldsymbol{M}_{t-1}) = p_\theta(\boldsymbol{a}_t | \boldsymbol{a}_{t-1}) p_\theta(\boldsymbol{M}_t | \boldsymbol{M}_{t-1}) \,. \tag{3}$$

This allows us to model the evolution of $\boldsymbol{a}_t$ and $\boldsymbol{M}_t$ separately.

The second part of the model represents how the HI pixels are generated from $\boldsymbol{a}_t$ and $\boldsymbol{M}_t$, which is given by

$$\boldsymbol{y}_t \sim p_\theta(\boldsymbol{y}_t | \boldsymbol{a}_t, \boldsymbol{M}_t) \,, \tag{4}$$

where $\boldsymbol{y}_t = [\boldsymbol{y}_{1,t}^\top, \ldots, \boldsymbol{y}_{N,t}^\top]^\top$ denotes the HI in lexicographic ordering. Note that the pixels $\boldsymbol{y}_t$ are assumed to be conditionally independent given $\boldsymbol{a}_t$ and $\boldsymbol{M}_t$. These PDFs are defined explicitly in Section IV.

Inference step: This step, which constitutes the solution to the MTHU problem, consists in computing the posterior PDF of the EMs and of the abundances given the observed pixels, which is given by:

$$\begin{aligned} &p_\theta(\boldsymbol{a}_1, \boldsymbol{M}_1, \ldots, \boldsymbol{a}_T, \boldsymbol{M}_T | \boldsymbol{y}_1, \ldots, \boldsymbol{y}_T) \qquad (5) \\ &= \frac{p_\theta(\boldsymbol{a}_0, \boldsymbol{M}_0) \prod_{t=1}^T p_\theta(\boldsymbol{y}_t | \boldsymbol{a}_t, \boldsymbol{M}_t) p_\theta(\boldsymbol{a}_t | \boldsymbol{a}_{t-1}) p_\theta(\boldsymbol{M}_t | \boldsymbol{M}_{t-1})}{p_\theta(\boldsymbol{y}_1, \ldots, \boldsymbol{y}_T)} \,, \end{aligned}$$

where the r.h.s. of (5) was obtained using the Bayes rule and the factorization in (2)–(4). The PDF in (5) generally does not

have a closed form solution [75]. One efficient solution is to use variational inference based on SGD [27], which attempts to find an approximate posterior $q \in \mathcal{Q}$ within a family of distributions $\mathcal{Q}$ that is as close as possible to the true posterior in (5). This approximation is often obtained by maximizing a lower bound on the marginal log-likelihood, the so-called *evidence lower bound* (ELBO):

$$\mathrm{ELBO}(q,\theta) \leqslant \log\, p_\theta(\boldsymbol{y}_1,\ldots,\boldsymbol{y}_T)\,, \tag{6}$$

see Section V for a detailed explanation. Under specific conditions over the model and posterior family $\mathcal{Q}$, such as assuming conditionally Gaussian distributions, the maximization $\max_{q\in\mathcal{Q}} \mathrm{ELBO}(q,\theta)$ can be solved locally using SGD techniques, which are computationally efficient when compared to solutions based on Monte Carlo sampling [27].

The flexibility provided by the family of posterior distributions $\mathcal{Q}$ is paramount for the performance of the strategy. Recent works have considered neural networks, parameterized by ϕ, to represent the approximate posterior. Thus, instead of searching for q in a (continuous) family of distributions $\mathcal{Q}$, we search for the parameters ϕ, such that the parameterized posterior, denoted by q_ϕ, maximizes the ELBO. Thus, the optimization becomes $\max_\phi \mathrm{ELBO}(q_\phi,\theta)$. For problems with a temporal Markov structure, RNNs provide a parameterization that, although flexible, is computationally efficient. Moreover, they explicitly explore the temporal structure in the data, having shown excellent performance in various sequence modeling tasks [28]. This will motivate us to use an RNN in the parametrization of our posterior in Section V-C.

Finally, the parameters of the generative model, θ, are also learned within the same framework. The underlying idea is to perform type-II maximum likelihood (ML) estimation [76], that is, $\max_\theta \log p_\theta(\boldsymbol{y}_1,\ldots,\boldsymbol{y}_T)$. However, since computing $p_\theta(\boldsymbol{y}_1,\ldots,\boldsymbol{y}_T)$ is intractable, θ is also computed by maximizing the ELBO w.r.t. θ using SGD. Thus, as $\mathrm{ELBO}(q_\phi,\theta)$ is maximized w.r.t. q_ϕ, the lower bound in (6) becomes tighter and its maximization w.r.t. θ better approximates ML estimation. Thus, the complete inference problem is formulated as the maximization of the ELBO w.r.t. both the posterior and the model parameters, that is, $\max_{\theta,\,\phi} \mathrm{ELBO}(q_\phi,\theta)$.

IV. Proposed model

The modeling step will be divided as follows. First, we develop a mixing model and represent EM variability (which defines $p_\theta(\boldsymbol{y}_t|\boldsymbol{a}_t,\boldsymbol{M}_t)$). Next, we consider the dynamical behavior of the EMs, and finally of the abundances (which define $p_\theta(\boldsymbol{M}_t|\boldsymbol{M}_{t-1})$ and $p_\theta(\boldsymbol{a}_t|\boldsymbol{a}_{t-1})$, respectively).

A. Mixture model with EM variability

As discussed in Section II, devising EM models that combine flexibility to represent complex spectral variability with simplicity of having a small number of parameters is challenging. Flexible models such as the PLMM [44], GLMM [46] and ALMM [47] have many degrees of freedom and require additional regularization strategies to guarantee physically meaningful solutions, while simpler models such as the ELMM [45] are too restrictive to represent complex spectral variability. An important information which can be used in the design of mixing models accounting for endmember variability is the spectral correlation of EM signatures [4]. This points to a natural representation of EMs as smooth functions. Although the smoothness of the EMs can be introduced through regularization (see, e.g., [77], [78]), this leads to high-dimensional and potentially costly HU solutions. On the other hand, a more efficient and interpretable model can be obtained by directly parametrizing smoothness using properly selected basis functions [79].

In this work, we consider an EM model inspired by the GLMM [46], which represents spectral variability using a multiplicative scaling of reference EM spectra that vary for each band, endmember and pixel. However, instead of using regularizations we propose to constrain the scaling factors to be linear combinations of spectrally smooth functions. The resulting model, which we call Smooth GLMM (SGLMM), represents each observed HI pixel $\boldsymbol{y}_{n,t}$ as follows:

$$\boldsymbol{y}_{n,t} = \underbrace{\left(\boldsymbol{M}_0 \odot (\mathbb{1} + \boldsymbol{D}\widetilde{\boldsymbol{\Psi}}_{n,t})\right)}_{\boldsymbol{M}_{n,t}} \boldsymbol{a}_{n,t} + \boldsymbol{r}_{n,t}\,, \tag{7}$$

where $\odot$ represents the Hadamard (elementwise) product, $\mathbb{1}$ is an $L \times P$ matrix of ones, $\boldsymbol{M}_0 \in \mathbb{R}^{L\times P}$ a set of reference or average EM signatures, matrix $\boldsymbol{D} \in \mathbb{R}^{L\times K}$ contains K spectrally smooth basis vectors as its columns, and $\widetilde{\boldsymbol{\Psi}}_{n,t} \in \mathbb{R}^{K\times P}$ contains the low-dimensional coefficients that parameterize the variability of each EM. Vector $\boldsymbol{r}_{n,t} \in \mathbb{R}^L$ denotes zero-mean additive Gaussian noise.

It is instructive to analyze how the variability of the endmembers $\boldsymbol{M}_{n,t}$ is introduced in the model (7). The EM matrix $\boldsymbol{M}_{n,t}$ is formed by scaling the reference EMs in $\boldsymbol{M}_0$ bandwise by matrix $\mathbb{1} + \boldsymbol{D}\widetilde{\boldsymbol{\Psi}}_{n,t} \in \mathbb{R}^{L\times P}$. This model is similar to the GLMM, the difference being in the structure of this multiplicative scaling matrix. First, note that when $\widetilde{\boldsymbol{\Psi}}_{n,t} \approx \boldsymbol{0}$, the term $\boldsymbol{D}\widetilde{\boldsymbol{\Psi}}_{n,t}$ is also small and the scaling factors will be close to $\mathbb{1}$, meaning that $\boldsymbol{M}_{n,t} \approx \boldsymbol{M}_0$, i.e., the spectral variability is small. Thus, the amount of EM variability depends directly on the amplitude of the elements of $\widetilde{\boldsymbol{\Psi}}_{n,t}$. Second, matrix $\boldsymbol{D}\widetilde{\boldsymbol{\Psi}}_{n,t}$ represents a perturbation over the constant scaling $\mathbb{1}$, and its properties depend directly on the choice of $\boldsymbol{D}$. Thus, by properly selecting $\boldsymbol{D}$ we can constrain $\boldsymbol{D}\widetilde{\boldsymbol{\Psi}}_{n,t}$ to represent smooth functions with few parameters, leading to smooth spectral variations in $\boldsymbol{M}_{n,t}$. Following an idea used in [79] for robust HU with smooth additive residual terms, we select the columns of $\boldsymbol{D}$ as the first K rows of the discrete cosine transform (DCT) matrix.

The SGLMM models endmember variability using KP parameters. The number of basis functions K gives a trade-off between existing models in the literature: when $K = 1$, $\boldsymbol{D}$ will contain only a constant vector and the model becomes equivalent to the ELMM [45], whereas for $K = L$ it has the same flexibility as the PLMM [44] and GLMM [46]. Values of $K \ll L$ should give the SGLMM sufficient flexibility to represent smooth spectral variability accurately.

Representing the spectral variability parameters in vectorized form as $\boldsymbol{\psi}_{n,t} = \mathrm{vec}(\widetilde{\boldsymbol{\Psi}}_{n,t})$ and assuming the noise $\boldsymbol{r}_{n,t}$

to be independent for each pixel, the PDFs in the generative model (4) can be rewritten equivalently in terms of $\boldsymbol{\psi}_{n,t}$ as

$$p_\theta(\boldsymbol{y}_t|\boldsymbol{a}_t,\boldsymbol{\psi}_t)=\prod_{n=1}^{N}p_\theta(\boldsymbol{y}_{n,t}|\boldsymbol{a}_{n,t},\boldsymbol{\psi}_{n,t})\,, \tag{8}$$

where

$$\begin{aligned}&p_\theta(\boldsymbol{y}_{n,t}|\boldsymbol{a}_{n,t},\boldsymbol{\psi}_{n,t})\\&\quad=\mathcal{N}\Big(\big(\boldsymbol{M}_0\odot(\mathbb{1}+\boldsymbol{D}\,\mathrm{vec}^{-1}(\boldsymbol{\psi}_{n,t}))\big)\boldsymbol{a}_{n,t},\,\sigma_r^2\boldsymbol{I}\Big)\,,\end{aligned} \tag{9}$$

in which $\sigma_r\in\mathbb{R}_+^*$ is the standard deviation of the measurement noise, which is assumed to be independent and identically distributed for different bands.

B. Dynamical model for the EM scaling parameters

In this work, we consider $\boldsymbol{M}_0$ to be a deterministic parameter of the model and estimated from the observed HIs using an approximate ML framework (i.e., $\boldsymbol{M}_0\in\theta$). This means that the EM matrix for each t and n, $\boldsymbol{M}_{n,t}=\boldsymbol{M}_0\odot(\mathbb{1}+\boldsymbol{D}\tilde{\boldsymbol{\Psi}}_{n,t})$, is a deterministic function of the lower-dimensional vector of scaling factors $\boldsymbol{\psi}_{n,t}$. Thus, we can substitute the problem of estimating the very high-dimensional $p_\theta(\boldsymbol{M}_t|\boldsymbol{M}_{t-1})$ by the problem of estimating $p_\theta(\boldsymbol{\psi}_t|\boldsymbol{\psi}_{t-1})$, with $\boldsymbol{\psi}_t=\left[\boldsymbol{\psi}_{1,t}^\top,\ldots,\boldsymbol{\psi}_{N,t}^\top\right]^\top$. Since vector $\boldsymbol{\psi}_t$ is still high dimensional, we consider an independence assumption on the time evolution between different pixels:

$$p_\theta(\boldsymbol{\psi}_t|\boldsymbol{\psi}_{t-1})=\prod_{n=1}^{N}p_\theta(\boldsymbol{\psi}_{n,t}|\boldsymbol{\psi}_{n,t-1})\,, \tag{10}$$

for $t\geqslant 1$, where the prior PDF for time instant $t=0$ will be specified later in Section IV-D. We consider a Gaussian distribution to represent the evolution of $\boldsymbol{\psi}_{n,t}$:

$$p_\theta(\boldsymbol{\psi}_{n,t}|\boldsymbol{\psi}_{n,t-1})=\mathcal{N}\big(\boldsymbol{\psi}_{n,t-1},\sigma_\psi^2\boldsymbol{I}_{PK}\big)\,, \tag{11}$$

where $\sigma_\psi\in\mathbb{R}_+^*$ is the distribution standard deviation, which controls its uncertainty and is assumed to be isotropic. Note that the evolution of $\boldsymbol{\psi}_{n,t}$ is not assumed to be affected by abrupt changes, which leads us to consider σ_ψ constant. This assumption is motivated from the fact that the reflectance of materials are primarily influenced by their physico-chemical composition (e.g., particle size and roughness in packed particle spectra [80], or biophysical parameters in leaf spectra [81]), which we assume to change smoothly at fine time scales.

C. Abundances model

In order to represent the abundances dynamical behavior, we first assume their time evolution to be independent for different pixels in order to make the problem tractable, that is,

$$p_\theta(\boldsymbol{a}_t|\boldsymbol{a}_{t-1})=\prod_{n=1}^{N}p_\theta(\boldsymbol{a}_{n,t}|\boldsymbol{a}_{n,t-1})\,, \tag{12}$$

for $t\geqslant 1$, where the prior PDF for time instant $t=0$ will be specified later in Section IV-D. To represent the time evolution at each pixel, we consider a Dirichlet distribution. The Dirichlet is a natural choice of distribution to model the abundances as it enforces the physical constraints that the elements of $\boldsymbol{a}_t$ should be nonnegativity and sum to one [82], [83]. The transition PDF is then given by

$$p_\theta(\boldsymbol{a}_{n,t}|\boldsymbol{a}_{n,t-1})=\mathrm{Dir}(\boldsymbol{\alpha}_{n,t})\,, \tag{13}$$

where $\boldsymbol{\alpha}_{n,t}\in\mathbb{R}_+^P$ denotes the concentration parameters, which are a function of the abundances at the previous time instant, $\boldsymbol{a}_{n,t-1}$ (i.e., the parameters of $p_\theta(\boldsymbol{a}_{n,t}|\boldsymbol{a}_{n,t-1})$, $\boldsymbol{\alpha}_{n,t}$, are a function of the conditioning variable). Note that the uncertainty of the abundances predictions is represented implicitly in $\boldsymbol{\alpha}_{n,t}$, where small concentration values yield low uncertainty and temporally smooth transition, whereas large values lead to higher uncertainty, allowing for more changes.

However, the Dirichlet distribution can make inference difficult. One workaround consists of using, e.g., Laplace's method, which approximates the Dirichlet distribution $\mathrm{Dir}(\boldsymbol{\alpha}_{n,t})$ by a Gaussian with mean and inverse covariance equal to the mode of the original distribution and the Hessian of its negative logarithm, respectively [23]. However, since the Dirichlet distribution is supported at the simplex, this approximation can be inaccurate. To overcome this problem, MacKay [26] proposed to perform this approximation in the so-called *softmax basis*, which consists in a mapping $\boldsymbol{\pi}^{-1}:\boldsymbol{a}_{n,t}\mapsto\boldsymbol{c}_{n,t}$ from the unity simplex to $\mathbb{R}^P$, where $\boldsymbol{\pi}$ is the softmax function:

$$\boldsymbol{\pi}^{-1}(\boldsymbol{a}_{n,t})=\boldsymbol{c}_{n,t}\,, \tag{14}$$

$$\pi_i(\boldsymbol{c}_{n,t})=\frac{\exp(c_{n,t,i})}{\sum_j\exp(c_{n,t,j})}\,,\quad i\in\{1,\ldots,P\}\,, \tag{15}$$

with π_i, $c_{n,t,i}$ being the i-th positions of $\boldsymbol{\pi}$ and $\boldsymbol{c}_{n,t}$. This approximation is very accurate and has been used in several works to facilitate statistical inference [84], [85]. Thus, replacing $\boldsymbol{a}_{n,t}$ by the softmax parameters $\boldsymbol{c}_{n,t}$, we achieve the following alternative representation of (13) [26]:

$$\begin{aligned}&p_\theta(\boldsymbol{c}_{n,t}|\boldsymbol{c}_{n,t-1})\\&\quad=\frac{\Gamma(\sum_{i=1}^P\alpha_{n,t,i})}{\prod_{i=1}^P\Gamma(\alpha_{n,t,i})}\prod_{i=1}^{P}\pi_i(\boldsymbol{c}_{n,t})^{\alpha_{n,t,i}}g(\mathbf{1}^\top\boldsymbol{c}_{n,t})\,,\end{aligned} \tag{16}$$

where $\alpha_{n,t,i}$ is the i-th position of $\boldsymbol{\alpha}_{n,t}$, Γ denotes the Gamma function, and g is an arbitrary distribution used to constrain an extra degree of freedom (since the Dirichlet has only $P-1$ degrees of freedom), selected as $g(x)\propto\exp(-\frac{\epsilon}{2}x^2)$ for mathematical convenience [26]. The Gaussian approximation of this distribution is then given by $p_\theta(\boldsymbol{c}_{n,t}|\boldsymbol{c}_{n,t-1})\approx\mathcal{N}(\boldsymbol{\mu}_{n,t},\boldsymbol{\Sigma}_{n,t})$ [26], [84], with $\boldsymbol{\mu}_{n,t}$ given by

$$\mu_{n,t,i}=\log(\alpha_{n,t,i})-\frac{1}{P}\sum_{\ell=1}^{P}\log(\alpha_{n,t,\ell})\,, \tag{17}$$

where $\mu_{n,t,i}$ is the i-th position of $\boldsymbol{\mu}_{n,t}$ and $\boldsymbol{\Sigma}_{n,t}$ is the negative Hessian of (16) at $\boldsymbol{c}_{n,t}=\boldsymbol{\mu}_{n,t}$. Note that the mean and covariance $\boldsymbol{\mu}_{n,t}$ and $\boldsymbol{\Sigma}_{n,t}$ are a function of $\boldsymbol{\alpha}_{n,t}$ and, consequently, depend implicitly on $\boldsymbol{c}_{n,t-1}$.

Therefore, we can approximate the transition PDF (13) by a Gaussian one on the softmax basis. Note that the relationship between the parameters of both models, that is, between $\boldsymbol{\alpha}_{n,t}$ and $\boldsymbol{\mu}_{n,t}$ and $\boldsymbol{\Sigma}_{n,t}$, is nonlinear and burdensome to compute. However, by working on the softmax basis we do not need

to specify $p_\theta(\boldsymbol{a}_{n,t}|\boldsymbol{a}_{n,t-1})$ explicitly in (13). Instead, we can directly define the Gaussian transition for $p_\theta(\boldsymbol{c}_{n,t}|\boldsymbol{c}_{n,t-1})$, which is mathematically more convenient. This implicitly defines a transition probability $p_\theta(\boldsymbol{a}_{n,t}|\boldsymbol{a}_{n,t-1})$ by mapping $\boldsymbol{c}_{n,t} \sim p_\theta(\boldsymbol{c}_{n,t}|\boldsymbol{c}_{n,t-1})$ into the simplex, approximating a Dirichlet distribution. Thus, we consider the following model:

$$p_\theta(\boldsymbol{c}_{n,t}|\boldsymbol{c}_{n,t-1}) = \mathcal{N}\big(\boldsymbol{c}_{n,t-1},\, \sigma_a^2(\boldsymbol{c}_{n,t-1})\boldsymbol{I}_P\big)\,, \tag{18}$$

where $\boldsymbol{I}_P$ is a $P \times P$ identity matrix. Note that, for simplicity, the covariance matrix in (18) was constrained to be isotropic, and is scaled by $\sigma_a^2(\boldsymbol{c}_{n,t-1})$. The function $\sigma_a : \mathbb{R}^P \to \mathbb{R}_+^*$ computes the standard deviation of each element of $\boldsymbol{c}_{n,t} \sim p_\theta(\boldsymbol{c}_{n,t}|\boldsymbol{c}_{n,t-1})$ as a function of $\boldsymbol{c}_{n,t-1}$. Thus, it directly influences the amount of change in the abundances: the larger $\sigma_a(\boldsymbol{c}_{n,t-1})$, the larger the changes we expect to observe between $\boldsymbol{c}_{n,t-1}$ and $\boldsymbol{c}_{n,t}$. This function, which is part of the generative model, will be learned during inference using a maximum likelihood approach. It will be parameterized using a fully connected neural network with R_{σ_a} layers, where each hidden layer has P neurons and uses the ReLU activation function, and the output layer maps to a scalar and uses an exponential activation function to ensure the output is positive.

D. The complete model

To finish the model derivation, we need to define the initial PDFs at time $t = 0$, which under the new parametrization of the abundances and endmembers, which we assume to be pixelwise independent Gaussian distributions, given by:

$$p_\theta(\boldsymbol{c}_0, \boldsymbol{\psi}_0) = \prod_{n=1}^{N} p_\theta(\boldsymbol{c}_{n,0}) p_\theta(\boldsymbol{\psi}_{n,0})\,, \tag{19}$$

$$p_\theta(\boldsymbol{c}_{n,0}) = \mathcal{N}\big(\boldsymbol{\nu}_0^c, \mathrm{diag}(\boldsymbol{\gamma}_0^c)^2\big)\,, \tag{20}$$

$$p_\theta(\boldsymbol{\psi}_{n,0}) = \mathcal{N}\big(\boldsymbol{\nu}_0^\psi, \mathrm{diag}(\boldsymbol{\gamma}_0^\psi)^2\big)\,, \tag{21}$$

for all $n = 1, \ldots, N$, where the means $\boldsymbol{\nu}_0^c$, $\boldsymbol{\nu}_0^\psi$ and diagonal covariance parameters, $\boldsymbol{\gamma}_0^c$, $\boldsymbol{\gamma}_0^\psi$ are constant and shared among all pixels in order to reduce the amount of parameters in the model. Finally, the measurement model (9) can be written using the softmax abundance reparametrization as:

$$\begin{aligned} &p_\theta(\boldsymbol{y}_{n,t}|\boldsymbol{c}_{n,t}, \boldsymbol{\psi}_{n,t}) \\ &= \mathcal{N}\Big(\big(\boldsymbol{M}_0 \odot (\mathbb{1} + \boldsymbol{D}\,\mathrm{vec}^{-1}(\boldsymbol{\psi}_{n,t}))\big)\boldsymbol{\pi}(\boldsymbol{c}_{n,t}),\, \sigma_r^2 \boldsymbol{I}\Big)\,, \end{aligned} \tag{22}$$

where $\boldsymbol{\pi}$ is the softmax function.

The final dynamical model is then given by equations (20), (21) (initial PDFs), (18), (11) (the dynamical model) and (22) (the measurement model). Finally, we denote the parameters of the model which will be estimated from the data using approximate ML inference by $\theta = \{\boldsymbol{M}_0, \sigma_r, \sigma_a, \boldsymbol{\nu}_0^c, \boldsymbol{\nu}_0^\psi, \boldsymbol{\gamma}_0^c, \boldsymbol{\gamma}_0^\psi\}$. An illustrative diagram of the proposed generative model can be seen in Figure 3.

V. Variational inference with RNNs for HU

In this section, we will present the proposed solution to the inference step, referred to as ReDSUNN. Considering the parametrization of the abundances and of endmember variability derived in the previous section, this task, which consists in performing MTHU, becomes that of approximating the posterior distribution:

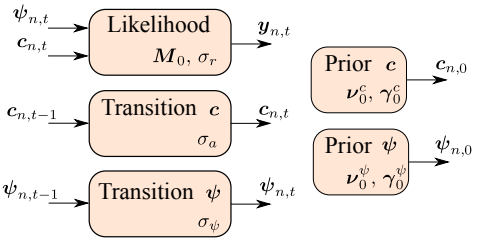

Fig. 3: Illustrative diagram of the proposed generative model.

$$p_\theta\big(\boldsymbol{c}_1, \boldsymbol{\psi}_1, \ldots, \boldsymbol{c}_T, \boldsymbol{\psi}_T | \boldsymbol{y}_1, \ldots, \boldsymbol{y}_T\big)\,, \tag{23}$$

where $\boldsymbol{c}_t = \big[\boldsymbol{c}_{1,t}^\top, \ldots, \boldsymbol{c}_{N,t}^\top\big]^\top$. First, let us denote with an underline the collection of variables at all time instants:

$$\underline{\boldsymbol{y}} = \{\boldsymbol{y}_1, \ldots, \boldsymbol{y}_T\}\,, \tag{24}$$

$$\underline{\boldsymbol{y}}_n = \{\boldsymbol{y}_{n,1}, \ldots, \boldsymbol{y}_{n,T}\}\,, \tag{25}$$

$$\underline{\boldsymbol{c}} = \{\boldsymbol{c}_0, \ldots, \boldsymbol{c}_T\}\,, \tag{26}$$

$$\underline{\boldsymbol{\psi}} = \{\boldsymbol{\psi}_0, \ldots, \boldsymbol{\psi}_T\}\,. \tag{27}$$

As discussed in Section III, due to the nonlinearity in the model caused by the interaction between the abundances and the variability scaling factors, and the potentially high dimensionality of these variables, it is not possible to compute the posterior distribution (23) in closed form. In this work, we adopt a deep variational inference framework: we consider a parametric surrogate distribution $q_\phi(\underline{\boldsymbol{c}}, \underline{\boldsymbol{\psi}}|\underline{\boldsymbol{y}})$ from a sufficiently flexible family with parameters ϕ, and learn its parameters by minimizing the Kullback-Leibler (KL) divergence between $q_\phi(\underline{\boldsymbol{c}}, \underline{\boldsymbol{\psi}}|\underline{\boldsymbol{y}})$ and the true posterior $p_\theta(\underline{\boldsymbol{c}}, \underline{\boldsymbol{\psi}}|\underline{\boldsymbol{y}})$:

$$\begin{aligned} \mathrm{KL}\left(p_\theta(\underline{\boldsymbol{c}}, \underline{\boldsymbol{\psi}}|\underline{\boldsymbol{y}}) \| q_\phi(\underline{\boldsymbol{c}}, \underline{\boldsymbol{\psi}}|\underline{\boldsymbol{y}})\right) &= \log p_\theta(\underline{\boldsymbol{y}}) \\ &+ \mathbb{E}_{q_\phi(\underline{\boldsymbol{c}}, \underline{\boldsymbol{\psi}}|\underline{\boldsymbol{y}})}\{\log q_\phi(\underline{\boldsymbol{c}}, \underline{\boldsymbol{\psi}}|\underline{\boldsymbol{y}})\} \\ &- \mathbb{E}_{q_\phi(\underline{\boldsymbol{c}}, \underline{\boldsymbol{\psi}}|\underline{\boldsymbol{y}})}\{\log p_\theta(\underline{\boldsymbol{c}}, \underline{\boldsymbol{\psi}}, \underline{\boldsymbol{y}})\}\,. \end{aligned} \tag{28}$$

Since the KL divergence is nonnegative and $\log p_\theta(\underline{\boldsymbol{y}})$ is a constant, the above expression can be equivalently minimized by maximizing a lower bound to the data likelihood formed by the last two terms in the right hand side of the expression, which is the so-called ELBO [23], [86]:

$$\begin{aligned} \log p_\theta(\underline{\boldsymbol{y}}) \geqslant\, &\mathbb{E}_{q_\phi(\underline{\boldsymbol{c}}, \underline{\boldsymbol{\psi}}|\underline{\boldsymbol{y}})}\{\log p_\theta(\underline{\boldsymbol{c}}, \underline{\boldsymbol{\psi}}, \underline{\boldsymbol{y}})\} \\ &- \mathbb{E}_{q_\phi(\underline{\boldsymbol{c}}, \underline{\boldsymbol{\psi}}|\underline{\boldsymbol{y}})}\{\log q_\phi(\underline{\boldsymbol{c}}, \underline{\boldsymbol{\psi}}|\underline{\boldsymbol{y}})\}\,. \end{aligned} \tag{29}$$

Recent advances in variational deep learning such as in variational autoencoders has made it possible to devise efficient algorithms to maximize (29) when the PDFs are possibly parameterized by deep neural network using, e.g., stochastic backpropagation algorithms [87]. Furthermore, the conditional independence assumptions of the model in the previous section can be exploited to further simplify the inference problem. In the following, we factorize the ELBO both in the temporal as well as in the pixel dimensions.

Note that, as discussed in Section III, (29) will be optimized both with respect to the parameters of the posterior distribution ϕ, but also with respect to the parameters of the generative

model $p_\theta(\underline{\boldsymbol{c}}, \underline{\boldsymbol{\psi}}, \underline{\boldsymbol{y}})$ in the set θ in order to estimate them by approximate ML inference. In the following, we will use the Markov and pixelwise conditional independence assumptions of the generative model in Section IV to factorize $q_\phi(\underline{\boldsymbol{c}}, \underline{\boldsymbol{\psi}}|\underline{\boldsymbol{y}})$ and simplify the solution to the inference problem.

A. Factorizing the posterior distribution

Factorizing the posterior distribution in time: Various kinds of parametrizations of the distribution $q_\phi(\underline{\boldsymbol{c}}, \underline{\boldsymbol{\psi}}|\underline{\boldsymbol{y}})$ have been proposed. One of the simplest is to consider a mean field assumption [88], which assumes that $\{\boldsymbol{c}_t, \boldsymbol{\psi}_t\}$ and $\{\boldsymbol{c}_{t'}, \boldsymbol{\psi}_{t'}\}$ are conditionally independent given $\underline{\boldsymbol{y}}$, for all $t \neq t'$. However, this disregards the temporal structure of the data. A more suitable factorization can be obtained by noting that the Markov property of the model can be used to show that the true posterior factorizes as

$$p_\theta(\underline{\boldsymbol{c}}, \underline{\boldsymbol{\psi}}|\underline{\boldsymbol{y}}) = p_\theta(\boldsymbol{c}_0, \boldsymbol{\psi}_0|\underline{\boldsymbol{y}}) \prod_{t=1}^{T} p_\theta(\boldsymbol{c}_t, \boldsymbol{\psi}_t|\boldsymbol{c}_{t-1}, \boldsymbol{\psi}_{t-1}, \underline{\boldsymbol{y}}) \,.$$

Incorporating this assumption into the variational approximation $q_\phi(\underline{\boldsymbol{a}}, \underline{\boldsymbol{\psi}}|\underline{\boldsymbol{y}})$ leads to a similar factorization:

$$q_\phi(\underline{\boldsymbol{c}}, \underline{\boldsymbol{\psi}}|\underline{\boldsymbol{y}}) = q_\phi(\boldsymbol{c}_0, \boldsymbol{\psi}_0|\underline{\boldsymbol{y}}) \prod_{t=1}^{T} q_\phi(\boldsymbol{c}_t, \boldsymbol{\psi}_t|\boldsymbol{c}_{t-1}, \boldsymbol{\psi}_{t-1}, \underline{\boldsymbol{y}}) \,, \tag{30}$$

which preserves the temporal dependency of the model.

Factorizing the posterior distribution in pixels: The vectors $\boldsymbol{c}_t, \boldsymbol{\psi}_t$ in (30) contain the abundances and variability coefficients for all image pixels, and are thus of very high dimension. Therefore, additional simplifications are necessary in order to make inference tractable. One important property is that in the model derived in Section IV, the initial, transition, and measurement PDFs (equations (20), (21), (18), (11) and (22)) can be factorized among the different image pixels. Thus, the inference process can be factorized at the pixel level, which leads to the following form for the posterior distribution:

$$\begin{aligned} &q_\phi(\boldsymbol{c}_t, \boldsymbol{\psi}_t|\boldsymbol{c}_{t-1}, \boldsymbol{\psi}_{t-1}, \underline{\boldsymbol{y}}) \\ &\quad = \prod_{n=1}^{N} q_\phi(\boldsymbol{c}_{n,t}, \boldsymbol{\psi}_{n,t}|\boldsymbol{c}_{n,t-1}, \boldsymbol{\psi}_{n,t-1}, \underline{\boldsymbol{y}}_n) \,, \end{aligned} \tag{31}$$

for $t \geqslant 1$, and similarly for the initial PDF at $t = 0$:

$$q_\phi(\boldsymbol{c}_0, \boldsymbol{\psi}_0|\underline{\boldsymbol{y}}) = \prod_{n=1}^{N} q_\phi(\boldsymbol{c}_{n,0}, \boldsymbol{\psi}_{n,0}|\underline{\boldsymbol{y}}_n) \,. \tag{32}$$

Although this factorization does not directly consider spatial correlation between different pixels, which has been found to be a useful source of prior information in HU [25], [82], it allows us to work with pixelwise variational posterior PDFs (i.e., the r.h.s. of (31) and (32)) which have a much lower dimension, thus reducing the computational burden associated with the inference step. To some extent, spatial information can still be introduced indirectly by constraining the parametrization of the variational posterior distributions among different pixels, which will be explained in the rest of this subsection. The incorporation of spatial information directly through the probabilistic model will be investigated in a future work.

Parameterizing the posterior: A key aspect of the model is how to parameterize the posterior PDFs of the different pixels in the r.h.s. of (31) and (32). First, variational inference implies selecting a parametric family of distributions from which to select q_ϕ, which directly impact the results. Note that the true posterior in (5) might have a complex form and be possibly multimodal, however, its form is not known in advance. Thus, as in recent works in deep variational inference (see, e.g., [86]) we considered a Gaussian family for q_ϕ since this will simplify the maximization of the ELBO considerably (through, e.g., the reparametrization trick and closed form expressions for KL divergences), leading to important computation savings. Thus, it can be expressed as:

$$\begin{aligned} &q_\phi\big(\boldsymbol{c}_{n,t}, \boldsymbol{\psi}_{n,t}\big|\boldsymbol{c}_{n,t-1}, \boldsymbol{\psi}_{n,t-1}, \underline{\boldsymbol{y}}_n\big) \\ &\qquad = \mathcal{N}\big(\boldsymbol{\mu}_\phi^{c,\psi}(\boldsymbol{\Upsilon}_{n,t}), \mathrm{diag}(\boldsymbol{\sigma}_\phi^{c,\psi}(\boldsymbol{\Upsilon}_{n,t}))^2\big) \,, \end{aligned} \tag{33}$$

where $\boldsymbol{\Upsilon}_{n,t} = \big\{\boldsymbol{c}_{n,t-1}, \boldsymbol{\psi}_{n,t-1}, \underline{\boldsymbol{y}}_n\big\}$ and $\boldsymbol{\mu}_\phi^{c,\psi}$ and $\boldsymbol{\sigma}_\phi^{c,\psi}$ are functions (e.g., neural networks parameterized by ϕ) which compute the parameters of the posterior distribution, mapping the data $\{\boldsymbol{c}_{n,t-1}, \boldsymbol{\psi}_{n,t-1}, \underline{\boldsymbol{y}}_n\}$ to the mean and the square root of the diagonal covariance matrix of the Gaussian posterior, respectively. For convenience of notation, we decompose $\boldsymbol{\mu}_\phi^{c,\psi}$ and $\boldsymbol{\sigma}_\phi^{c,\psi}$ into two functions:

$$\boldsymbol{\mu}_\phi^{c,\psi} = \begin{bmatrix} \boldsymbol{\mu}_\phi^{c} \\ \boldsymbol{\mu}_\phi^{\psi} \end{bmatrix} \,, \qquad \boldsymbol{\sigma}_\phi^{c,\psi} = \begin{bmatrix} \boldsymbol{\sigma}_\phi^{c} \\ \boldsymbol{\sigma}_\phi^{\psi} \end{bmatrix} \,. \tag{34}$$

Note that functions $\boldsymbol{\mu}_\phi^{c}$ and $\boldsymbol{\sigma}_\phi^{c}$ compute the mean and square root of the diagonal covariance matrix of the $q_\phi\big(\boldsymbol{c}_{n,t}\big|\boldsymbol{c}_{n,t-1}, \boldsymbol{\psi}_{n,t-1}, \underline{\boldsymbol{y}}_n\big)$, while $\boldsymbol{\mu}_\phi^{\psi}$ and $\boldsymbol{\sigma}_\phi^{\psi}$ compute the mean and square root of the diagonal covariance matrix of $q_\phi\big(\boldsymbol{\psi}_{n,t}\big|\boldsymbol{c}_{n,t-1}, \boldsymbol{\psi}_{n,t-1}, \underline{\boldsymbol{y}}_n\big)$.

A Gaussian parametrization is also used for the posterior distribution of the initial PDF:

$$q_\phi(\boldsymbol{c}_0, \boldsymbol{\psi}_0|\underline{\boldsymbol{y}}) = \mathcal{N}\big(\boldsymbol{\zeta}^{c,\psi}, \mathrm{diag}(\boldsymbol{\xi}^{c,\psi})^2\big) \,, \tag{35}$$

where a fixed distribution was used for all pixels, with $\boldsymbol{\zeta}^{c,\psi}$ and $\boldsymbol{\xi}^{c,\psi}$ being the mean and the diagonal of the square root of the covariance matrix, respectively.

An important observation is that we consider a *shared parametrization*, where the posterior in (33) and (35) has the same form for all pixels. More precisely, this means that the same functions $\boldsymbol{\mu}_\phi^{c,\psi}$ and $\boldsymbol{\sigma}_\phi^{c,\psi}$ are used to compute the posterior mean and covariance for every HI pixel, given the input data $\boldsymbol{\Upsilon}_{n,t}$. This is an important characteristic of the method, since it allows information from different pixels (i.e., from the whole image) to be leveraged jointly in the estimation of the model and, consequently, of the abundances and variability coefficients in each pixel, $\boldsymbol{c}_{n,t}, \boldsymbol{\psi}_{n,t}$, $n = 1, \ldots, N$.

B. Factorizing the ELBO cost function

Using the simplifications derived in the previous subsection, in the following we will rewrite the ELBO cost function (29) in terms of the factorized model.

Factorizing the ELBO temporally: Using the factorization (30) and the Markovity of the model, the lower bound in (29) can be written as [88]:

$$\log p_\theta(\underline{\boldsymbol{y}}) \geqslant \mathcal{L}(\theta,\phi,\underline{\boldsymbol{y}}) = \sum_{t=1}^{T} \mathbb{E}_{q_\phi(\boldsymbol{c}_t,\boldsymbol{\psi}_t|\underline{\boldsymbol{y}})}\big\{\log p_\theta(\boldsymbol{y}_t|\boldsymbol{c}_t,\boldsymbol{\psi}_t)\big\}$$
$$- \mathrm{KL}\left(q_\phi(\boldsymbol{c}_0,\boldsymbol{\psi}_0|\underline{\boldsymbol{y}})\|p_\theta(\boldsymbol{a}_0,\boldsymbol{\psi}_0)\right) - \sum_{t=1}^{T} \mathbb{E}_{q_\phi(\boldsymbol{c}_{t-1},\boldsymbol{\psi}_{t-1}|\underline{\boldsymbol{y}})}\Big\{$$
$$\mathrm{KL}\left(q_\phi(\boldsymbol{c}_t,\boldsymbol{\psi}_t|\boldsymbol{c}_{t-1},\boldsymbol{\psi}_{t-1},\underline{\boldsymbol{y}})\|p_\theta(\boldsymbol{c}_t,\boldsymbol{\psi}_t|\boldsymbol{c}_{t-1},\boldsymbol{\psi}_{t-1})\right)\Big\}. \tag{36}$$

Factorizing at pixel level: Using the pixelwise factorization of the generative and posterior PDFs discussed in Section IV, we can simplify each term of (36). To this end, we use the fact that $\mathrm{KL}(p(x_1,x_2)\|q(x_1,x_2)) = \mathrm{KL}(p(x_1)\|q(x_1)) + \mathrm{KL}(p(x_2)\|q(x_2))$ when both $p(x_1,x_2) = p(x_1)p(x_2)$ and $q(x_1,x_2) = q(x_1)q(x_2)$ are independent, and the fact that $\mathbb{E}_{p(x_1,x_2)}\{f(x_1)\} = \mathbb{E}_{p(x_1)}\{f(x_1)\}$. We proceed to analyse each term of (36) in the following.

First term: Using the factorization of the measurement model in (8), the first term in the r.h.s. of (36) becomes:

$$\mathbb{E}_{q_\phi(\boldsymbol{c}_t,\boldsymbol{\psi}_t|\underline{\boldsymbol{y}})}\big\{\log p_\theta(\boldsymbol{y}_t|\boldsymbol{c}_t,\boldsymbol{\psi}_t)\big\}$$
$$= \sum_{n=1}^{N} \mathbb{E}_{q_\phi(\boldsymbol{c}_{n,t},\boldsymbol{\psi}_{n,t}|\underline{\boldsymbol{y}}_n)}\big\{\log p_\theta(\boldsymbol{y}_{n,t}|\boldsymbol{c}_{n,t},\boldsymbol{\psi}_{n,t})\big\}. \tag{37}$$

Second term: Using the pixelwise independence of the initial PDF in the generative model (19) and in the posterior (32), the KL divergence can be written as:

$$\mathrm{KL}\big(q_\phi(\boldsymbol{c}_0,\boldsymbol{\psi}_0|\underline{\boldsymbol{y}})\|p_\theta(\boldsymbol{c}_0,\boldsymbol{\psi}_0)\big)$$
$$= \sum_{n=1}^{N} \mathrm{KL}\left(q_\phi(\boldsymbol{c}_{n,0},\boldsymbol{\psi}_{n,0}|\underline{\boldsymbol{y}}_n)\|p_\theta(\boldsymbol{c}_{n,0},\boldsymbol{\psi}_{n,0})\right). \tag{38}$$

Third term: Using the pixelwise independence of the posterior (31) and of the predictive PDFs (12), (10), this term can be written as:

$$\mathbb{E}_{q_\phi(\boldsymbol{c}_{t-1},\boldsymbol{\psi}_{t-1}|\underline{\boldsymbol{y}})}\big\{\mathrm{KL}\big($$
$$q_\phi(\boldsymbol{c}_t,\boldsymbol{\psi}_t|\boldsymbol{c}_{t-1},\boldsymbol{\psi}_{t-1},\underline{\boldsymbol{y}})\|p_\theta(\boldsymbol{c}_t,\boldsymbol{\psi}_t|\boldsymbol{c}_{t-1},\boldsymbol{\psi}_{t-1})\big)\big\}$$
$$= \sum_{n=1}^{N} \mathbb{E}_{q_\phi(\boldsymbol{c}_{n,t-1},\boldsymbol{\psi}_{n,t-1}|\underline{\boldsymbol{y}}_n)}\big\{\mathrm{KL}\big(\tag{39}$$
$$q_\phi(\boldsymbol{c}_{n,t},\boldsymbol{\psi}_{n,t}|\boldsymbol{c}_{n,t-1},\boldsymbol{\psi}_{n,t-1},\underline{\boldsymbol{y}}_n)\|p_\theta(\boldsymbol{c}_{n,t},\boldsymbol{\psi}_{n,t}|\boldsymbol{c}_{t-1},\boldsymbol{\psi}_{t-1})\big)\big\}.$$

Combining these results, we can write the cost function $\mathcal{L}(\theta,\phi,\underline{\boldsymbol{y}})$ as in equation (40) (depicted on top of the next page). Details on the computation of the log-likelihood and KL divergences can be found in Appendix A.

C. An RNN-based implementation

A key question is how to define the functions $\boldsymbol{\mu}_\phi^c$, $\boldsymbol{\mu}_\phi^\psi$, $\boldsymbol{\sigma}_\phi^c$ and $\boldsymbol{\sigma}_\phi^\psi$ in (33) and (34), which parameterize the approximate posterior distribution. On the one hand, these have to be flexible to be able to approximate the true posterior, which cannot be written in the form of a simple and well-known distribution. On the other hand, it is important to incorporate information from a physical modeling of the problem to make inference process more efficient, interpretable and stable. Thus, we will parameterize the variational posterior distribution using a lightweight RNN and, whenever possible, leveraging physically motivated models.

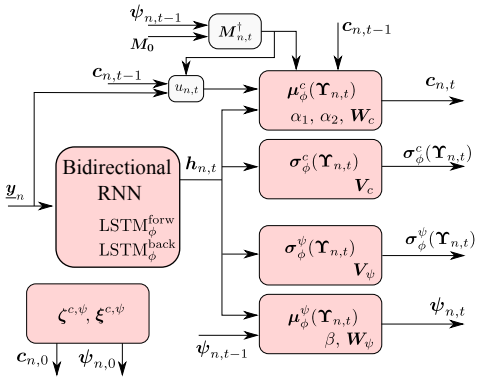

Fig. 4: Diagram of the proposed network implementing the posteriors $q_\phi(\boldsymbol{c}_{n,t},\boldsymbol{\psi}_{n,t}|\boldsymbol{c}_{n,t-1},\boldsymbol{\psi}_{n,t-1},\underline{\boldsymbol{y}}_n)$ and $q_\phi(\boldsymbol{c}_0,\boldsymbol{\psi}_0|\underline{\boldsymbol{y}})$.

First, a bidirectional RNN is used to compute a set of feature-based representations, denoted by $\boldsymbol{h}_{n,t} \in \mathbb{R}^H$, from the image pixel sequence $\underline{\boldsymbol{y}}_n$. In particular, we compute $\boldsymbol{h}_{n,t}$ by combining the hidden states learned by two LSTMs [89]:

$$\boldsymbol{h}_{n,t}^{\mathrm{forw}} = \mathrm{LSTM}_\phi^{\mathrm{forw}}(\boldsymbol{h}_{n,t-1}^{\mathrm{forw}},\boldsymbol{y}_{n,t}),\ \ t=1,\dots,T, \tag{41}$$
$$\boldsymbol{h}_{n,t}^{\mathrm{back}} = \mathrm{LSTM}_\phi^{\mathrm{back}}(\boldsymbol{h}_{n,t+1}^{\mathrm{back}},\boldsymbol{y}_{n,t}),\ \ t=T,\dots,1, \tag{42}$$
$$\boldsymbol{h}_{n,t} = \frac{1}{2}\big(\boldsymbol{h}_{n,t}^{\mathrm{forw}} + \boldsymbol{h}_{n,t}^{\mathrm{back}}\big), \tag{43}$$

for $n = 1,\dots,N$, where $\mathrm{LSTM}_\phi^{\mathrm{forw}}$ and $\mathrm{LSTM}_\phi^{\mathrm{back}}$ denote two LSTMs which process the data forward and backwards in time, respectively; their hidden state representation being given by $\boldsymbol{h}_{n,t}^{\mathrm{forw}}$ and $\boldsymbol{h}_{n,t}^{\mathrm{back}}$. We choose LSTMs due to their excellent performance in various sequence modeling tasks [28]. Moreover, a bidirectional RNN (i.e., two LSTMs) is used because at every time instant the posterior in (33) depends on the HI pixels at all time instants, $\underline{\boldsymbol{y}}_n$, whereas the LSTMs in (41) and (42) depend only on past and future data, respectively.

The dimension of the RNN representation is selected as $H = (K+1)P$ (i.e., the dimension of the state vector). The gating units of the LSTMs use the sigmoid nonlinearity, while the input and hidden state units use the uses the hyperbolic tangent nonlinearity. Note that the parameters of these LSTMs will also be learned during inference by SDG using backpropagation through time [28].

We now use the representation $\boldsymbol{h}_{n,t}$ to parameterize $\boldsymbol{\mu}_\phi^c$, $\boldsymbol{\mu}_\phi^\psi$, $\boldsymbol{\sigma}_\phi^c$ and $\boldsymbol{\sigma}_\phi^\psi$. To introduce physical knowledge, we follow the general idea of using hybrid models [22], [62], in which an approximate model is complemented by a learnable component (in this case derived from the RNN). In particular, for the posterior mean of the abundances, $\boldsymbol{\mu}_\phi^c$, we construct an approximate model by assuming that 1) a least squares solution provides a crude abundance estimate, 2) the abundances are temporally smooth but may undergo sudden changes, and 3) abrupt abundance changes lead to abrupt changes in the pixels. For the variability parameters, $\boldsymbol{\mu}_\phi^\psi$, we consider it to

$$\mathcal{L}(\theta,\phi,\underline{\boldsymbol{y}}) = \sum_{t=1}^{T}\sum_{n=1}^{N}\mathbb{E}_{q_\phi(\boldsymbol{c}_{n,t},\boldsymbol{\psi}_{n,t}|\underline{\boldsymbol{y}}_n)}\big\{\log p_\theta(\boldsymbol{y}_{n,t}|\boldsymbol{c}_{n,t},\boldsymbol{\psi}_{n,t})\big\} - \sum_{n=1}^{N}\mathrm{KL}\left(q_\phi(\boldsymbol{c}_{n,0},\boldsymbol{\psi}_{n,0}|\underline{\boldsymbol{y}}_n)\|p_\theta(\boldsymbol{c}_{n,0},\boldsymbol{\psi}_{n,0})\right)$$
$$- \sum_{t=1}^{T}\sum_{n=1}^{N}\mathbb{E}_{q_\phi(\boldsymbol{c}_{n,t-1},\boldsymbol{\psi}_{n,t-1}|\underline{\boldsymbol{y}}_n)}\Big\{\mathrm{KL}\left(q_\phi(\boldsymbol{c}_{n,t},\boldsymbol{\psi}_{n,t}|\boldsymbol{c}_{n,t-1},\boldsymbol{\psi}_{n,t-1},\underline{\boldsymbol{y}}_n)\|p_\theta(\boldsymbol{c}_{n,t},\boldsymbol{\psi}_{n,t}|\boldsymbol{c}_{n,t-1},\boldsymbol{\psi}_{n,t-1})\right)\Big\}. \quad (40)$$

be temporally smooth. For the standard deviations, $\boldsymbol{\sigma}_\phi^c$ and $\boldsymbol{\sigma}_\phi^\psi$, we don't have a good physical model; thus, we use a purely non-parametric representation. In the following, we define each of these functions explicitly; an illustrative diagram can be seen in Figure 4.

Considering the RNN features $\boldsymbol{h}_{n,t}$, the abundance posterior means $\boldsymbol{\mu}_\phi^c$ is then parameterized as:

$$\boldsymbol{\mu}_\phi^c(\boldsymbol{\Upsilon}_{n,t}) = \boldsymbol{\pi}^{-1}\Big(\alpha_1(1-u_{n,t})\boldsymbol{\pi}(\boldsymbol{c}_{n,t-1}) + \alpha_2 u_{n,t}\big(\boldsymbol{M}_{n,t-1}^\dagger\boldsymbol{y}_{n,t} + \boldsymbol{W}_c\boldsymbol{h}_{n,t}\big)\Big), \quad (44)$$

where $\boldsymbol{M}_{n,t-1} = \boldsymbol{M}_0 \odot (\mathbb{1} + \boldsymbol{D}\,\mathrm{vec}^{-1}(\boldsymbol{\psi}_{n,t-1}))$ is the predicted EM matrix at pixel n and time $t-1$, α_1 and α_2 are trainable, real-valued weighting coefficients, and $\boldsymbol{W}_c \in \mathbb{R}^{P\times(K+1)P}$ is a trainable matrix that maps the hidden RNN representations to the abundances in the softmax basis. The scalar coefficient $u_{n,t} \in [0,1]$, defined as $u_{n,t} = \frac{1}{2P}\|s(\boldsymbol{M}_{n,t-1}^\dagger\boldsymbol{y}_{n,t}) - \boldsymbol{\pi}(\boldsymbol{c}_{n,t-1})\|_1$, measures the difference between the predicted abundances $\boldsymbol{\pi}(\boldsymbol{c}_{n,t-1})$ and a crude estimation of the current abundances at time t, given by $s(\boldsymbol{M}_{n,t-1}^\dagger\boldsymbol{y}_{n,t})$, where the fixed function $s(\cdot)$ projects the linear regression solution $\boldsymbol{M}_{n,t-1}^\dagger\boldsymbol{y}_{n,t}$ to the unit simplex. Thus, $u_{n,t}$ works as a crude abrupt change detector.

The parametrization (44) can be seen as a weighted combination of three terms: the abundances at the previous time instant, a crude abundance estimate at time t computed by linear regression, and a non-parametric term depending on $\boldsymbol{h}_{n,t}$. The balance between them depends on the trainable weights and on the change detector $u_{n,t}$. When there are no changes, $u_{n,t}$ is small, which gives a higher contribution to the predicted abundances $\boldsymbol{c}_{n,t-1}$ in (44). On the other hand, if there is an abrupt change, $u_{n,t}$ is large, giving a higher contribution to the sum of the last two terms in (44), which is a linear regression-based abundance estimate augmented by a non-parametric RNN-based representation. This parametrization is particularly relevant since the generative model (18) does not explicitly represent abrupt changes.

For the function $\boldsymbol{\mu}_\phi^\psi$, to leverage temporal smoothness we consider a weighted linear combination of the variability coefficients at the previous time instant $\boldsymbol{\psi}_{n,t-1}$ and a linear mapping of the hidden RNN representation:

$$\boldsymbol{\mu}_\phi^\psi(\boldsymbol{\Upsilon}_{n,t}) = \beta\boldsymbol{\psi}_{n,t-1} + \boldsymbol{W}_\psi\boldsymbol{h}_{n,t}\,, \quad (45)$$

where β is a real-valued weight, and $\boldsymbol{W}_\psi \in \mathbb{R}^{KP\times(K+1)P}$ is a matrix that computes the variability coefficients' innovation from the RNN representation $\boldsymbol{h}_{n,t}$, both of which are trainable. Note that by not considering abrupt changes to occur in $\boldsymbol{\psi}_t$ we obtain a simpler model compared to $\boldsymbol{\mu}_\phi^c$.

The standard deviations $\boldsymbol{\sigma}_\phi^c$ and $\boldsymbol{\sigma}_\phi^\psi$ are computed based on a fully non-parametric model, which is given as linear mappings of the RNN representations $\boldsymbol{h}_{n,t}$:

$$\boldsymbol{\sigma}_\phi^c(\boldsymbol{\Upsilon}_{n,t}) = \exp\left(\boldsymbol{V}_c\boldsymbol{h}_{n,t}\right), \quad (46)$$
$$\boldsymbol{\sigma}_\phi^\psi(\boldsymbol{\Upsilon}_{n,t}) = \exp\left(\boldsymbol{V}_\psi\boldsymbol{h}_{n,t}\right), \quad (47)$$

where $\boldsymbol{V}_c \in \mathbb{R}^{P\times(K+1)P}$ and $\boldsymbol{V}_\psi \in \mathbb{R}^{KP\times(K+1)P}$ are the transformation matrices, and the exponential function is applied elementwise in order to ensure nonnegativity of the standard deviations. The parameters of the approximate posterior are finally denoted by $\phi = \big\{\boldsymbol{\zeta}^{c,\psi}, \boldsymbol{\xi}^{c,\psi}, \mathrm{LSTM}_\phi^{\text{forw}}, \mathrm{LSTM}_\phi^{\text{back}}, \alpha_1, \alpha_2, \beta, \boldsymbol{W}_c, \boldsymbol{V}_c, \boldsymbol{W}_\psi, \boldsymbol{V}_\psi\big\}$. Note that all parameters in ϕ will be learned using SGD.

Approximating the expectations and optimization: To optimize (40) using stochastic backpropagation, it is necessary to estimate gradients of expectations whose distribution depend on θ and ϕ, which are the parameters to be optimized. To address this issue, we consider the reparametrization trick, which provides low-variance gradient estimates [86]. This is performed by writing the random variables inside the expectations as deterministic functions of a random variable that does not depend on ϕ. In general, for a distribution $q_\phi(\boldsymbol{x})$ and function f, this can be formulated as $\mathbb{E}_{q_\phi(\boldsymbol{x})}\{f(\boldsymbol{x})\} = \mathbb{E}_{p(\boldsymbol{\epsilon})}\{f(g(\boldsymbol{\epsilon}))\}$, where g is a function such that $\boldsymbol{x}$ and $g(\boldsymbol{\epsilon})$ have the same distribution, and $p(\boldsymbol{\epsilon})$ does not depend on ϕ. Applying this to the expectations in (40) and considering that the posterior in our model is Gaussian, this is achieved as:

$$\left[\boldsymbol{a}_{n,t}^\top, \boldsymbol{\psi}_{n,t}^\top\right]^\top = \boldsymbol{\mu}_\phi^{c,\psi}(\boldsymbol{\Upsilon}_{n,t}) + \boldsymbol{\sigma}_\phi^{c,\psi}(\boldsymbol{\Upsilon}_{n,t}) \odot \boldsymbol{\epsilon}\,, \quad (48)$$

for $t = 1,\ldots,T$, where $\boldsymbol{\epsilon} \sim \mathcal{N}(\boldsymbol{0},\boldsymbol{I})$. It can be verified that the random variables in (48) are sampled according to the distribution $q_\phi(\boldsymbol{c}_{n,t},\boldsymbol{\psi}_{n,t}|\underline{\boldsymbol{y}}_n)$. Thus, by using this reparametrization, the expectations in (40) can be rewritten in terms of expectations of $\boldsymbol{\epsilon}$, which we subsequently approximate using a one-sample Monte Carlo estimate and denote by $\widehat{\mathcal{L}}(\theta,\phi,\underline{\boldsymbol{y}})$.

The approximated cost function $\widehat{\mathcal{L}}(\theta,\phi,\underline{\boldsymbol{y}})$ is then optimized with respect to both θ and ϕ (i.e., the parameters of the generative model and of the variational posterior) using the Adam stochastic optimization method [87]. We used a learning rate of 0.001 and a batch size of 128. Training was performed for 30 epochs. The full MTSU process performed by ReDSUNN is summarized in Algorithm 1.

Since the cost function is non-convex, the initialization of the parameters can have an important impact on the solution. The parameters of the neural networks σ_a, $\mathrm{LSTM}_\phi^{\text{forw}}$, $\mathrm{LSTM}_\phi^{\text{back}}$, and the matrices $\boldsymbol{V}_c$ and $\boldsymbol{V}_\psi$ are initialized randomly using Glorot initialization [90]. $\boldsymbol{W}_c$ and $\boldsymbol{W}_\psi$ are initialized with zeros, and $\beta = \alpha_1 = \alpha_2 = 1$. $\boldsymbol{M}_0$ was initialized using the vertex component analysis (VCA) algorithm [91], and $\sigma_r = 0.0001$ (corresponding to an SNR of about 35dB

Algorithm 1: ReDSUNN

Input : HIs $\boldsymbol{y}_1, \ldots, \boldsymbol{y}_T$, hyperparameters P, K and σ_ψ.

Initialize θ_0 and ϕ_0 as described in Section V-C ;

Use Adam [87] to maximize $\widehat{\mathcal{L}}(\theta, \phi, \underline{\boldsymbol{y}})$ w.r.t. both ϕ and θ ;

for $n = 1, \ldots, N$ **do**

  Compute $\widehat{\boldsymbol{c}}_{n,0}, \widehat{\boldsymbol{\psi}}_{n,0}$ as the means of $q_\phi(\boldsymbol{c}_{n,0}, \boldsymbol{\psi}_{n,0}|\underline{\boldsymbol{y}}_n)$ using (35);

  Compute the $\boldsymbol{h}_{n,1}, \ldots, \boldsymbol{h}_{n,T}$ using (41), (42) and (43) ;

  for $t = 1, \ldots, T$ **do**

    Compute $\widehat{\boldsymbol{c}}_{n,t}, \widehat{\boldsymbol{\psi}}_{n,t}$ as the means of $q_\phi(\boldsymbol{c}_t, \boldsymbol{\psi}_t|\widehat{\boldsymbol{c}}_{t-1}, \widehat{\boldsymbol{\psi}}_{t-1}, \underline{\boldsymbol{y}})$ using (44) and (45) ;

  end

end

Set $\widehat{\boldsymbol{a}}_{n,t} = \boldsymbol{\pi}(\widehat{\boldsymbol{c}}_{n,t})$, $\widehat{\boldsymbol{M}}_{n,t} = \widehat{\boldsymbol{M}}_0 \odot (\mathbb{1} + \boldsymbol{D}\,\mathrm{vec}^{-1}(\widehat{\boldsymbol{\psi}}_{n,t}))$;

Output: $\widehat{\boldsymbol{a}}_{n,t}$, $\widehat{\boldsymbol{M}}_{n,t}$, for $n = 1, \ldots, N$ and $t = 1, \ldots, T$.

for spectra with standard deviation 0.5). For the parameters of the initial prior and variational posterior PDFs, we initialized the means $\boldsymbol{\nu}_0^a$, $\boldsymbol{\nu}_0^\psi$ and $\boldsymbol{\zeta}^{c,\psi}$ with zeros, and the variances $\boldsymbol{\gamma}_0^a$, $\boldsymbol{\gamma}_0^\psi$ and $\boldsymbol{\xi}^{c,\psi}$ with ones, making the initial PDFs standard Gaussians.

D. Model complexity and comparisons

We now summarize the parameters of the generative model, of the variational posterior, and their dimensionality (i.e., the number of parameters that have to be inferred). This can be seen in Table I. To compute the number of parameters corresponding to the LSTMs, we note that each LSTM has four input-hidden weight matrices, four hidden-hidden weight matrices, and four biases (where the input is of size L, and the hidden state of size $(K+1)P$). It is instructive to compare the amount of parameters to other methods in the literature. By using a Markovity assumption, a shared posterior distribution for all pixels, and an RNN posterior parametrization, the amount of parameters to be learned by ReDSUNN in Table I does not scale with either N or T, differently from previous methods such as OU [7] or the HBUN [9].

TABLE I: Variables to be estimated and number of parameters.

Generative model (θ)	
$\boldsymbol{M}_0$	LP
σ_r	1
σ_a	$P(P+1)R_{\sigma_a}$
$\boldsymbol{\nu}_0^a, \boldsymbol{\nu}_0^\psi, \boldsymbol{\gamma}_0^a, \boldsymbol{\gamma}_0^\psi$	$2(K+1)P$
Variational posterior (ϕ)	
$\mathrm{LSTM}_\phi^{\mathrm{forw}}$, $\mathrm{LSTM}_\phi^{\mathrm{back}}$	$8(K+1)P\big((K+1)P + L + 1\big)$
α_1, α_2, β	3
$\boldsymbol{W}_c$, $\boldsymbol{V}_c$	$2(P^2(K+1))$
$\boldsymbol{W}_\psi$, $\boldsymbol{V}_\psi$	$2(KP^2(K+1))$
$\boldsymbol{\zeta}^{c,\psi}$, $\boldsymbol{\xi}^{c,\psi}$	$2(K+1)P$

VI. Results

The performance of the proposed ReDSUNN algorithm is evaluated using simulations with synthetic and real data. We compare our method with the fully constrained least squares (FCLS), online unmixing (OU) [7], HBUN [9], and with a Kalman filter and expectation maximization-based strategy (referred to simply as *Kalman*) [8]. The EMs used by FCLS were extracted by the VCA algorithm at each time instant [91]. The reference EMs required by the Kalman method, and the initialization for the EMs in OU, HBUN and for the proposed method were all set with the signatures obtained by applying VCA to the matrix $[\boldsymbol{y}_{1,1}, \ldots, \boldsymbol{y}_{n,t} \ldots, \boldsymbol{y}_{N,T}] \in \mathbb{R}^{L \times NT}$ formed by concatenating the HI pixels for all time instants.

TABLE II: Quantitative results of the simulations using synthetic data.

	$\mathrm{NRMSE}_{\boldsymbol{A}}$	$\mathrm{NRMSE}_{\boldsymbol{M}}$	$\mathrm{SAM}_{\boldsymbol{M}}$	$\mathrm{NRMSE}_{\boldsymbol{Y}}$	Time
Data Sequence 1 – DS1					
FCLS	0.537	–	–	0.086	**2.7**
OU	0.434	0.342	0.260	0.051	24.9
HBUN	0.479	0.355	0.162	**0.050**	542.6
Kalman	0.356	0.124	0.076	0.061	2422.8
ReDSUNN	**0.318**	**0.117**	**0.075**	0.089	479.0
Data Sequence 2 – DS2					
FCLS	0.500	–	–	0.122	**7.3**
OU	0.335	0.256	**0.120**	0.055	60.6
HBUN	0.474	0.515	0.141	**0.050**	2166.0
Kalman	0.659	12.222	0.496	0.108	5937.4
ReDSUNN	**0.294**	**0.203**	0.289	0.160	1231.3

The abundances and EM scaling factors estimated by ReDSUNN are set according to Algorithm 1. The hyperparameters of all algorithms were adjusted so as to obtain high abundance reconstruction performance. For ReDSUNN, parameters K and σ_ψ (which are not optimized) were searched within the ranges $K \in \{1, \ldots, 10\}$ and $\sigma_\psi \in \{10^{-5}, \ldots, 0.1, 1\}$, and $R_{\sigma_a} = 2$ layers were used to parameterize function $\sigma_a(\cdot)$ in (18). For the other algorithms, their parameters were selected in the ranges indicated in their original publications. The proposed method was implemented in Pytorch (codes will be available at https://github.com/ricardoborsoi/ReDSUNN). The remaining methods were implemented in Matlab (codes were provided by the original authors). All experiments were run in a desktop computer with an Intel Xeon™ W-2104 CPU with four 3.2GHz cores and 24GB of RAM. No GPU was used in the simulations. ReDSUNN, OU and Kalman used parallelization in their implementations.

The quantitative performance of the algorithms was evaluated using the average normalized mean squared error (NRMSE), between the abundances, EMs, and reconstructed HIs, which are computed as $\mathrm{NRMSE}_{\boldsymbol{A}} = \big(\frac{1}{T}\sum_{t=1}^{T}\sum_{n=1}^{N}\|\boldsymbol{a}_{n,t} - \widehat{\boldsymbol{a}}_{n,t}\|^2/\|\boldsymbol{a}_t\|^2\big)^{1/2}$, $\mathrm{NRMSE}_{\boldsymbol{M}} = \big(\frac{1}{NT}\sum_{t=1}^{T}\sum_{n=1}^{N}\|\boldsymbol{M}_{n,t} - \widehat{\boldsymbol{M}}_{n,t}\|_F^2/\|\boldsymbol{M}_{n,t}\|_F^2\big)^{1/2}$, and $\mathrm{NRMSE}_{\boldsymbol{Y}} = \big(\frac{1}{T}\sum_{t=1}^{T}\sum_{n=1}^{N}\big\|\boldsymbol{y}_{n,t} - \widehat{\boldsymbol{M}}_{n,t}\widehat{\boldsymbol{a}}_t\big\|^2/\|\boldsymbol{y}_t\|^2\big)^{1/2}$, where $\widehat{\boldsymbol{a}}_{n,t}$ and $\widehat{\boldsymbol{M}}_{n,t}$ denote the estimated abundances and EMs. To evaluate the EMs, we also computed the average spectral angle mapper (SAM) as $\mathrm{SAM}_{\boldsymbol{M}} = \frac{1}{TNP}\sum_{t=1}^{T}\sum_{n=1}^{N}\sum_{j=1}^{P}\arccos\Big(\frac{\boldsymbol{m}_{n,t,j}^\top\widehat{\boldsymbol{m}}_{n,t,j}}{\|\boldsymbol{m}_{n,t,j}\|\|\widehat{\boldsymbol{m}}_{n,t,j}\|}\Big)$, in which $\boldsymbol{m}_{n,t,j}$ and $\widehat{\boldsymbol{m}}_{n,t,j}$ are the true and estimated EM signatures for time t, pixel n and EM j.

A. Simulations with synthetic data

Two synthetic datasets were considered with spatiotemporal abundance and EM variability. The first dataset, referred to as

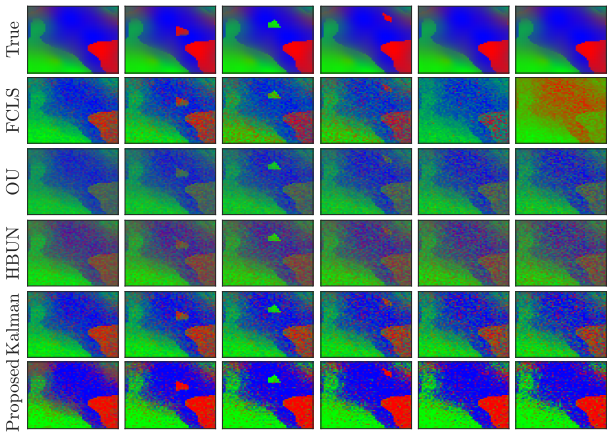

Fig. 5: Estimated abundance maps and ground truth for DS1, shown as composite color maps (that is, the abundances of the EMs #1, #2 and #3 are represented as the red, green, and blue color channels).

Data Sequence 1 (DS1), contains $P = 3$ EMs and $T = 6$ HIs. The HIs are generated from sequences of abundance maps with $N = 50 \times 50$ pixels containing localized abrupt changes for $t \in \{2, 3, 4, 5\}$ (depicted in the first row of Figure 5). The EMs for each pixel and time instant were generated as follows. First, three signatures with $L = 224$ bands were selected from the USGS library and used as a reference EM matrix. Then, for the first time instant ($t = 1$), spatial EM variability was introduced by following the model in [44], in which the EMs in each pixel ($\boldsymbol{M}_{n,1}$, $n = 1, \ldots, N$) were generated by multiplying the reference signatures with piecewise linear random scaling factors with amplitude in the interval $[0.85, 1.15]$. For each subsequent time instant $t > 1$, the EMs were also generated as scaled versions of the reference spectral signatures. However, to introduce temporal EM variability, the scaling factors at time t are defined to be the sum of the scaling factors at time $t - 1$ plus random piecewise linear functions in the range $[-0.1, 0.1]$. Samples of the generated EMs can be seen in Figure 6. These EM matrices $\boldsymbol{M}_{n,t}$ are then used to generate the HI pixels using the LMM (1), with the measurement noise $\boldsymbol{r}_{n,t}$ being white and Gaussian with an SNR of 30 dB. The second dataset, referred to as Data Sequence 2 (DS2), contained $P = 4$ EMs and $N = 50 \times 50$ pixels. A sequence of abundance maps generated randomly according to a Gaussian random field and containing small, spatially compact abrupt changes was considered to generate $T = 15$ HIs. To introduce realistic spectral variability, the EM signatures at each pixel and time instant were randomly selected from a set of pure pixels of water, vegetation, soil and road that were manually extracted from the Jasper Ridge HI, with $L = 198$ bands. The HI sequence was then generated according to the multitemporal LMM (1), with the $\boldsymbol{r}_{n,t}$ being white Gaussian noise with an SNR of 30 dB. The parameters of the ReDSUNN were $K = 10$ and $\sigma_\psi = 10^{-5}$ for DS1, and $K = 2$ and $\sigma_\psi = 10^{-5}$ for DS2. The quantitative results are presented in Table II, while the visual results (only shown for DS1 due to space limitations) are depicted in Figures 5 and 7.

1) Discussion: It can be seen from Table II that ReDSUNN achieved the best abundance estimation performance for both datasets. OU and HBUN achieved consistent but intermediate

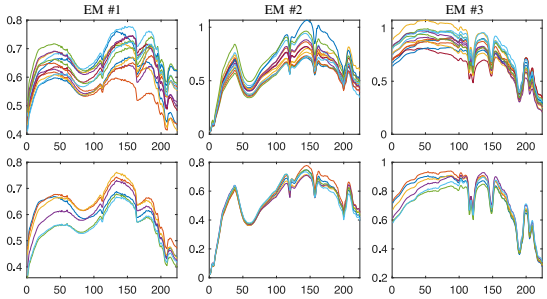

Fig. 6: True EMs for the DS1, sampled over space, for time instant $t = 3$ (top), and over time, for pixel $n = 1$ (bottom).

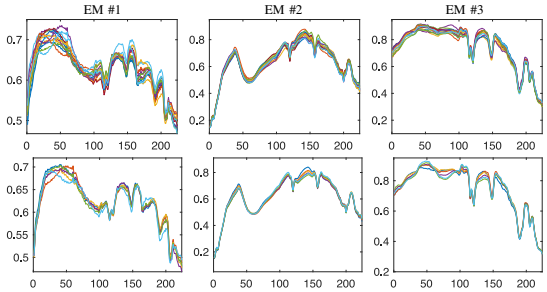

Fig. 7: Estimated EMs for the DS1, sampled over space, for time instant $t = 3$ (top), and over time, for pixel $n = 1$ (bottom).

results, while the performance of the Kalman filter was good for DS1 but very poor for DS2. The FLCS, which does not take temporal information of spatial EM variability into account, did not perform very well, having the worse abundance reconstructions on average for both datasets. From the estimated abundances in Figure 5, it can be seen that ReDSUNN's results are the closest to the ground truth. However, the results for all methods were relatively noisy. The abundances recovered by the Kalman filter, OU and HBUN indicated more heavily mixed pixels. FCLS achieves reasonable performance for $t \leqslant 4$, but led to a completely wrong estimation for $t = 6$. The changes occurring in the ground truth abundances can be observed in the estimations of all methods, although they are more clearly visible in the Kalman and ReDSUNN results since these methods led to a larger separation between the different materials. The visual abundance results for DS2 (not shown due to space limitations) were qualitatively similar to those of DS1, with the exception that the Kalman filter failed to identify the soil EM for all images in the sequence, which explains its poor performance.

The ReDSUNN method also obtained the best EM estimation performance all metrics except for the SAM in DS2, in which OU achieved the best result followed by HBUN. The Kalman filter obtained good results for DS1 (close to ReDSUNN), but poor results in DS2. This happened despite the Kalman filter obtaining reasonable image reconstruction errors $\text{NRMSE}_{\boldsymbol{Y}}$ for both DS1 and DS2. Samples of the true and estimated EMs in Figures 6 and 7 (only shown for DS1 and for ReDSUNN due to space limitations) indicate that the EMs are correctly recovered. However, there are some differences, particularly in the shape of the first EM (which show higher amplitude in the ground truth compared to the estimates). Moreover, the amount of variability was lower in

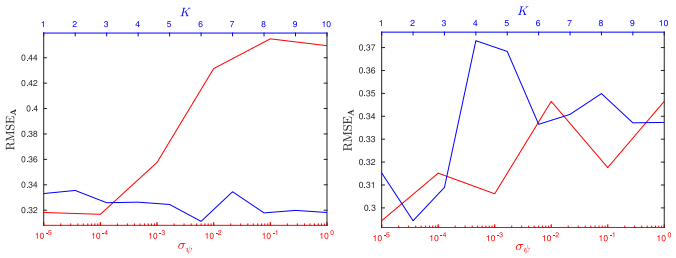

Fig. 8: Abundance RMSE as a function of hyperparameters K and σ_ψ for DS1 (left) and DS2 (right).

the retrieved EMs compared to the ground truth; this occurs for the synthetic examples since the hyperparameter σ_ψ, which controls the flexibility of the EM model, was selected to provide the best performance in terms of NRMSE$_{\boldsymbol{A}}$, leading to relatively small σ_ψ values. This is further illustrated in the experiments shown in Section VI-B.

The lowest image reconstruction errors (NRMSE$_{\boldsymbol{Y}}$) were obtained by OU and HBUN, while those obtained by ReDSUNN were similar to those by FCLS. This is expected, since NRMSE$_{\boldsymbol{Y}}$ is closely related to the number of learnable parameters of each algorithm, and is not directly related to the abundance or EM reconstruction performance. This explains the higher reconstruction error by ReDSUNN since, as discussed in Section V-D, the shared parametrization of the variational posterior PDF leads to a relatively low number of parameters, which also helps to mitigate overfitting. Nonetheless, for the synthetic data sequences (DS1 and DS2) ReDSUNN still has between 30% and 50% more learnable parameters than OU. Its parametrizaition becomes significantly more favorable when the images have a larger amounts of pixels, such as in the experiments with the Lake Tahoe images presented in Section VI-C. The computation times show a clear separation between FCLS and OU, which were faster, and HBUN, the Kalman filter and ReDSUNN, which took longer to run. This indicates that the proposed method has a competitive computational performance when compared to more complex algorithms.

B. Sensitivity analysis

To measure the influence of different hyperparameters on the performance of the method, we evaluated how NRMSE$_{\boldsymbol{A}}$ varied as a function of the hyperparameters, namely, the number of basis vectors for the variability model, K, and the innovation standard deviation of the EM variability parameters, σ_ψ. The results for both DS1 and DS2 can be seen in Figure 8. It can be seen that for DS1, the performance of ReDSUNN is not heavily affected by the number of bases K within the evaluated range. However, σ_ψ has a larger impact on the result, with smaller values leading to a lower NRMSE$_{\boldsymbol{A}}$. For DS2, smaller values for both parameters generally lead to lower NRMSE$_{\boldsymbol{A}}$ results, although there performance varied more with K and σ_ψ. For both datasets, small variations of these parameters around the optimal values lead to similar results. In general, the larger the value of σ_ψ, the more temporal EM variability is allowed by the model, whereas the larger the value of K, the more complex the spatial and temporal EM variability the model can represent. Devising a methodology to automatically tune these parameters is an interesting question for future work.

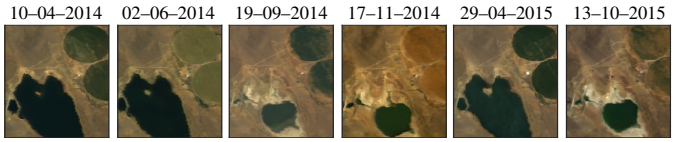

Fig. 9: True color depiction of the Lake Tahoe HIs and their acquisition dates.

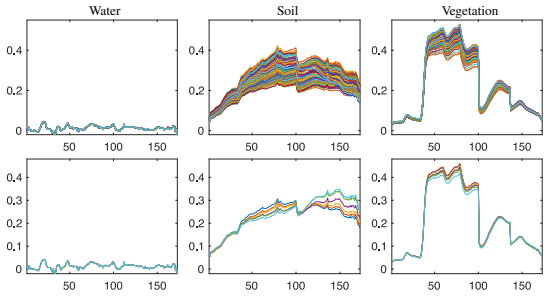

Fig. 10: Estimated EMs for the Lake Tahoe HIs, sampled over space, for time instant $t = 3$ (top), and over time, for pixel $n = 1$ (bottom).

TABLE III: Quantitative results for the Lake Tahoe HI sequence.

	FCLS	OU	HBUN	Kalman	ReDSUNN
NRMSE$_{\boldsymbol{Y}}$	0.321	0.058	**0.054**	0.185	0.114
Time	**16.1**	92.5	3381.9	4607.7	2857.2

C. Simulations with real data

To evaluate the performance of the algorithms on real data, we considered the Lake Tahoe HI sequence, which was originally described in [7]. It consists of a sequence of $T = 6$ images acquired over the Lake Tahoe area by the AVIRIS instruments, which are depicted in true color in Figure 9. Each HI contained $N = 16500$ pixels, and $L = 173$ bands were left after the removal of low-SNR and water absorption bands. This scene contains $P = 3$ predominant EMs, consisting of soil, water and vegetation, and considerable changes on the lake and on the crop circles can be observed between the images. The parameters of the ReDSUNN were set as $K = 3$, $\sigma_\psi = 1$, while the parameters of OU, HBUN and of the Kalman filter were selected as described in their original publications. The recovered abundances are depicted in Figure 11, while the recovered EMs (only shown for ReDSUNN due to space limitations) are shown in Figure 10. The reconstruction errors and the processing times are presented in Table III.

1) Discussion: From Figure 11, it can be seen that the FCLS method did not achieve a good performance in general, particularly for the fifth image in which there was a considerable confusion between the soil and vegetation EMs. The remaining algorithms achieve more stable performance due to taking the temporal information into account. The OU and the HBUN algorithms (both of which use the PLMM [44] model to represent the temporal endmember variability) behaved similarly to each other. Although these methods performed more stably than the FCLS, they still presented considerable

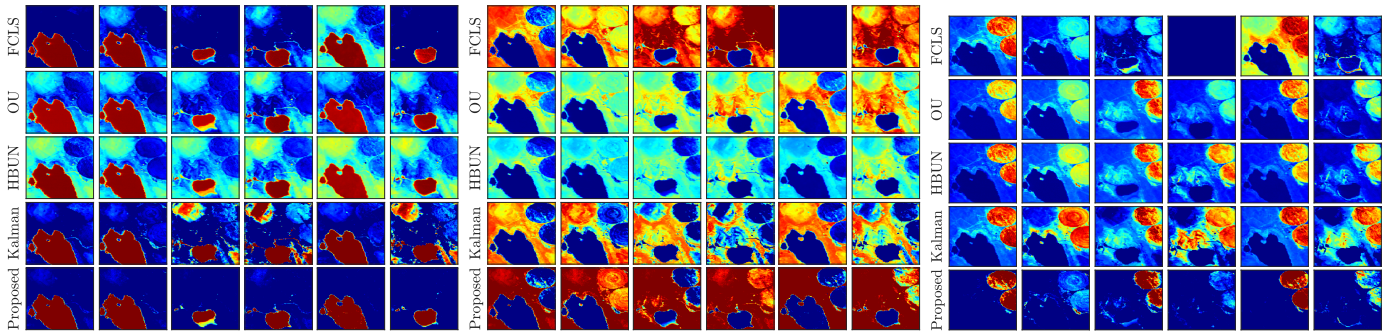

Fig. 11: Estimated abundances for the water (left panel), soil (central panel) and vegetation (right panel) EMs of the Lake Tahoe HIs.

water abundances outside of the lake region, which are predominantly composed by soil. The performance of the Kalman filter method was relatively poor for the third, fourth and sixth images (in which the area of the lake is small), and contained a considerable amount of artifacts. This happens since the Kalman filter method assumes the abundances to be constant over time when estimating the EMs. Consequently, it is not able to handle large abundance changes in the image sequence. The abundances estimated by ReDSUNN, on the other hand, showed a clear separation between the different materials, adequately capturing the abundance changes occurring in the HIs. Moreover, larger concentrations of mixed pixels were observed in regions that are meaningful, such as at the drying edge of the lake in the third image, and in some parts of the crop circles. The EMs recovered by ReDSUNN show considerable variability in soil and vegetation spectral, particularly over space, while the water spectra shows little variability. Moreover, the variability of the EMs can be spectrally localized, which can be observed most clearly in the temporal signatures of soil. Moreover, spatial EM variability was more significant than temporal EM variability. Note that the EM variability was also more significant in this example compared to the experiments with synthetic data since a larger value for the hyperparameter σ_ψ was selected.

The results in Table III show that HBUN and OU obtain the smallest reconstruction errors (NRMSE$_{\boldsymbol{Y}}$), while those obtained by Kalman and ReDSUNN, which have less learnable parameters, were larger, with FCLS having the largest NRMSE$_{\boldsymbol{Y}}$. The ratio between the computation times were similar to the synthetic examples, with the Kalman filter being the slowest and the OU the fastest among the MTHU methods that account for temporal information, and the proposed ReDSUNN method achieving intermediate results. This indicates that the methods scale similarly with the image size. Nevertheless, developing more efficient MTHU algorithms is an interesting subject for future work.

VII. Conclusions

This paper proposed a multitemporal hyperspectral unmixing method based on a variational recurrent neural network. A low-dimensional, dynamical state space model was presented to represent the spatial and temporal variations of the endmember spectra by expanding it over a small set of spectrally smooth basis vectors. The dynamics of the abundances were modelled using a Dirichlet distribution, which was approximated as a Gaussian in the softmax basis in order to improve the efficiency of the inference process. Based on this generative model, variational inference was considered to perform unmixing by approximating the posterior distribution of the abundances and endmembers. The Markov and independence properties of the model were also used to improve the efficiency of the solution. The posterior distribution was parameterized using a combination of a simple, physically interpretable, model and LSTM recurrent neural networks to improve flexibility while maintaining the physical interpretability of the abundances. In the proposed framework, all parameters were computed using stochastic backpropagation. Experimental results indicate that the proposed algorithm achieves better unmixing performance when compared to state-of-the-art methods, at a similar computational complexity, using both synthetic and real datasets.

Appendix A
Computing the terms in (40)

Due to the (conditionally) Gaussian assumptions in the generative model and in the variational posterior, the three terms inside the expectations in (40) can be computed analytically. The first term in (40) is the log-likelihood of a Gaussian PDF, which can be computed from (22). The second and third terms are KL divergences between Gaussian PDFs, which can be computed using the general result for two Gaussians of dimension D, given by $\mathrm{KL}\left(\mathcal{N}(\boldsymbol{\mu}_1,\boldsymbol{\Sigma}_1)\|\mathcal{N}(\boldsymbol{\mu}_2,\boldsymbol{\Sigma}_2)\right) = \frac{1}{2}\big(\log\frac{|\boldsymbol{\Sigma}_2|}{|\boldsymbol{\Sigma}_1|} - D + \mathrm{tr}\{\boldsymbol{\Sigma}_2^{-1}\boldsymbol{\Sigma}_1\} + (\boldsymbol{\mu}_2-\boldsymbol{\mu}_1)^\top\boldsymbol{\Sigma}_2^{-1}(\boldsymbol{\mu}_2-\boldsymbol{\mu}_1)\big)$. The second (resp., third) term are thus computed substituting the mean and covariance from (35) and (20), (21) (resp., (33) and (18), (11)) in the expression above by using the fact that $\boldsymbol{c}_{n,t}$ and $\boldsymbol{\psi}_{n,t}$ are independent in the generative model. For more details, see, e.g., [92].

References

[1] 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 Geoscience and Remote Sensing Magazine*, vol. 1, no. 2, pp. 6–36, 2013.

[2] G. A. Shaw and H.-h. K. Burke, "Spectral imaging for remote sensing," *Lincoln laboratory journal*, vol. 14, no. 1, pp. 3–28, 2003.

[3] N. Keshava and J. F. Mustard, "Spectral unmixing," *IEEE Signal Processing Magazine*, vol. 19, no. 1, pp. 44–57, 2002.

[4] R. A. Borsoi, T. Imbiriba, J. C. M. Bermudez, C. Richard, J. Chanussot, L. Drumetz, J.-Y. Tourneret, A. Zare, and C. Jutten, "Spectral variability in hyperspectral data unmixing: A comprehensive review," *IEEE Geoscience and Remote Sensing Magazine*, vol. 9, pp. 223–270, 2021.

[5] A. Zare and K. C. Ho, "Endmember variability in hyperspectral analysis: Addressing spectral variability during spectral unmixing," *IEEE Signal Processing Magazine*, vol. 31, pp. 95–104, January 2014.

[6] S. Henrot, J. Chanussot, and C. Jutten, "Dynamical spectral unmixing of multitemporal hyperspectral images," *IEEE Trans. Image Process.*, vol. 25, no. 7, pp. 3219–3232, 2016.

[7] P.-A. Thouvenin, N. Dobigeon, and J.-Y. Tourneret, "Online unmixing of multitemporal hyperspectral images accounting for spectral variability," *IEEE Trans. Image Process.*, vol. 25, no. 9, pp. 3979–3990, 2016.

[8] R. A. Borsoi, T. Imbiriba, P. Closas, J. C. M. Bermudez, and C. Richard, "Kalman filtering and expectation maximization for multitemporal spectral unmixing," *IEEE Geoscience and Remote Sensing Letters*, 2020.

[9] H. Liu, Y. Lu, Z. Wu, Q. Du, J. Chanussot, and Z. Wei, "Bayesian unmixing of hyperspectral image sequence with composite priors for abundance and endmember variability," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 60, pp. 1–15, 2021.

[10] B. Somers and G. P. Asner, "Invasive species mapping in hawaiian rainforests using multi-temporal hyperion spaceborne imaging spectroscopy," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 6, no. 2, pp. 351–359, 2013.

[11] ——, "Multi-temporal hyperspectral mixture analysis and feature selection for invasive species mapping in rainforests," *Remote Sensing of Environment*, vol. 136, pp. 14–27, 2013.

[12] C. L. Lippitt, D. A. Stow, D. A. Roberts, and L. L. Coulter, "Multidate MESMA for monitoring vegetation growth forms in southern california shrublands," *International journal of remote sensing*, vol. 39, no. 3, pp. 655–683, 2018.

[13] M. A. Goenaga, M. C. Torres-Madronero, M. Velez-Reyes, S. J. Van Bloem, and J. D. Chinea, "Unmixing analysis of a time series of hyperion images over the guánica dry forest in puerto rico," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 6, no. 2, pp. 329–338, 2013.

[14] S. Liu, D. Marinelli, L. Bruzzone, and F. Bovolo, "A review of change detection in multitemporal hyperspectral images: Current techniques, applications, and challenges," *IEEE Geoscience and Remote Sensing Magazine*, vol. 7, no. 2, pp. 140–158, 2019.

[15] Q. Guo, J. Zhang, C. Zhong, and Y. Zhang, "Change detection for hyperspectral images via convolutional sparse analysis and temporal spectral unmixing," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 14, pp. 4417–4426, 2021.

[16] R. A. Borsoi, T. Imbiriba, J. C. M. Bermudez, and C. Richard, "Fast unmixing and change detection in multitemporal hyperspectral data," *IEEE Transactions on Computational Imaging*, vol. 7, pp. 975–988, 2021.

[17] J. S. Bhatt and M. V. Joshi, "Deep learning in hyperspectral unmixing: A review," in *Proc. IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*. IEEE, 2020, pp. 2189–2192.

[18] B. Palsson, J. R. Sveinsson, and M. O. Ulfarsson, "Blind hyperspectral unmixing using autoencoders: A critical comparison," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 15, pp. 1340–1372, 2022.

[19] C. Zhou and M. R. Rodrigues, "ADMM-Based hyperspectral unmixing networks for abundance and endmember estimation," *IEEE Transactions on Geoscience and Remote Sensing*, 2021.

[20] M. Wang, M. Zhao, J. Chen, and S. Rahardja, "Nonlinear unmixing of hyperspectral data via deep autoencoder networks," *IEEE Geoscience and Remote Sensing Letters*, vol. 16, no. 9, pp. 1467–1471, 2019.

[21] R. A. Borsoi, T. Imbiriba, and J. C. M. Bermudez, "Deep generative endmember modeling: An application to unsupervised spectral unmixing," *IEEE Transactions on Computational Imaging*, vol. 6, pp. 374–384, 2019.

[22] T. Imbiriba, A. Demirkaya, J. Duník, O. Straka, D. Erdoğmuş, and P. Closas, "Hybrid Neural Network Augmented Physics-based Models for Nonlinear Filtering," in *Proc. FUSION conference*, Linkoping, Sweden, 2022.

[23] D. Barber, *Bayesian Reasoning and Machine Learning*. Cambridge University Press, 2012.

[24] J. T. Kent and K. V. Mardia, "Spatial classification using fuzzy membership models," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 10, no. 5, pp. 659–671, 1988.

[25] O. Eches, N. Dobigeon, and J.-Y. Tourneret, "Enhancing hyperspectral image unmixing with spatial correlations," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 49, no. 11, pp. 4239–4247, 2011.

[26] D. J. MacKay, "Choice of basis for Laplace approximation," *Machine learning*, vol. 33, no. 1, pp. 77–86, 1998.

[27] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," in *Proc. 2nd International Conference on Learning Representations (ICLR)*, Y. Bengio and Y. LeCun, Eds., Banff, AB, Canada, 2014.

[28] Z. C. Lipton, J. Berkowitz, and C. Elkan, "A critical review of recurrent neural networks for sequence learning," *arXiv preprint arXiv:1506.00019*, 2015.

[29] J. Sigurdsson, M. O. Ulfarsson, J. R. Sveinsson, and J. Bioucas-Dias, "Sparse distributed multitemporal hyperspectral unmixing," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 11, pp. 6069–6084, 2017.

[30] P.-A. Thouvenin, N. Dobigeon, and J.-Y. Tourneret, "A hierarchical bayesian model accounting for endmember variability and abrupt spectral changes to unmix multitemporal hyperspectral images," *IEEE Trans. Comput. Imaging*, vol. 4, no. 1, pp. 32–45, 2018.

[31] R. Zhuo, Y. Fang, L. Xu, Y. Chen, Y. Wang, and J. Peng, "A novel spectral-temporal Bayesian unmixing algorithm with spatial prior for Sentinel-2 time series," *Remote Sensing Letters*, vol. 13, no. 5, pp. 522–532, 2022.

[32] D. A. Roberts, M. Gardner, R. Church, S. Ustin, G. Scheer, and R. O. Green, "Mapping chaparral in the santa monica mountains using multiple endmember spectral mixture models," *Remote Sensing of Environment*, vol. 65, no. 3, pp. 267–279, 1998.

[33] K. L. Dudley, P. E. Dennison, K. L. Roth, D. A. Roberts, and A. R. Coates, "A multi-temporal spectral library approach for mapping vegetation species across spatial and temporal phenological gradients," *Remote Sensing of Environment*, vol. 167, pp. 121–134, 2015.

[34] Q. Wang, X. Ding, X. Tong, and P. M. Atkinson, "Spatio-temporal spectral unmixing of time-series images," *Remote Sensing of Environment*, vol. 259, p. 112407, 2021.

[35] ——, "Real-time spatiotemporal spectral unmixing of MODIS images," *IEEE Transactions on Geoscience and Remote Sensing*, 2021.

[36] B. Somers, G. P. Asner, L. Tits, and P. Coppin, "Endmember variability in spectral mixture analysis: A review," *Remote Sensing of Environment*, vol. 115, no. 7, pp. 1603–1616, 2011.

[37] M.-D. Iordache, J. M. Bioucas-Dias, and A. Plaza, "Sparse unmixing of hyperspectral data," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 49, no. 6, pp. 2014–2039, 2011.

[38] R. A. Borsoi, T. Imbiriba, J. C. M. Bermudez, and C. Richard, "A fast multiscale spatial regularization for sparse hyperspectral unmixing," *IEEE Geoscience and Remote Sensing Letters*, vol. 16, no. 4, pp. 598–602, April 2019.

[39] L. Drumetz, T. R. Meyer, J. Chanussot, A. L. Bertozzi, and C. Jutten, "Hyperspectral image unmixing with endmember bundles and group sparsity inducing mixed norms," *IEEE Transactions on Image Processing*, vol. 28, no. 7, pp. 3435–3450, 2019.

[40] T. Uezato, M. Fauvel, and N. Dobigeon, "Hyperspectral unmixing with spectral variability using adaptive bundles and double sparsity," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 57, no. 6, pp. 3980–3992, 2019.

[41] A. Halimi, N. Dobigeon, and J.-Y. Tourneret, "Unsupervised unmixing of hyperspectral images accounting for endmember variability," *IEEE Transactions on Image Processing*, vol. 24, no. 12, pp. 4904–4917, 2015.

[42] Y. Zhou, A. Rangarajan, and P. D. Gader, "A gaussian mixture model representation of endmember variability in hyperspectral unmixing," *IEEE Transactions on Image Processing*, vol. 27, no. 5, pp. 2242–2256, May 2018.

[43] X. Du, A. Zare, P. Gader, and D. Dranishnikov, "Spatial and spectral unmixing using the beta compositional model," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 7, no. 6, pp. 1994–2003, 2014.

[44] P.-A. Thouvenin, N. Dobigeon, and J.-Y. Tourneret, "Hyperspectral unmixing with spectral variability using a perturbed linear mixing model," *IEEE Trans. Signal Processing*, vol. 64, no. 2, pp. 525–538, Feb. 2016.

[45] L. Drumetz, M.-A. Veganzones, S. Henrot, R. Phlypo, J. Chanussot, and C. Jutten, "Blind hyperspectral unmixing using an extended linear mixing model to address spectral variability," *IEEE Trans. Image Process.*, vol. 25, no. 8, pp. 3890–3905, 2016.

[46] T. Imbiriba, R. A. Borsoi, and J. C. M. Bermudez, "Generalized linear mixing model accounting for endmember variability," in *Proc. IEEE ICASSP*, Calgary, Canada, 2018, pp. 1862–1866.

[47] D. Hong, N. Yokoya, J. Chanussot, and X. X. Zhu, "An augmented linear mixing model to address spectral variability for hyperspectral unmixing," *IEEE Transactions on Image Processing*, vol. 28, no. 4, pp. 1923–1938, 2019.

[48] R. A. Borsoi, T. Imbiriba, and J. C. Moreira Bermudez, "A data dependent multiscale model for hyperspectral unmixing with spectral variability," *IEEE Transactions on Image Processing*, vol. 29, pp. 3638–3651, 2020.

[49] T. Imbiriba, R. A. Borsoi, and J. C. M. Bermudez, "Low-rank tensor modeling for hyperspectral unmixing accounting for spectral variability," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 58, no. 3, pp. 1833–1842, 2020.

[50] T. Uezato, N. Yokoya, and W. He, "Illumination invariant hyperspectral image unmixing based on a digital surface model," *IEEE Transactions on Image Processing*, vol. 29, pp. 3652–3664, 2020.

[51] K. J. Guilfoyle, M. L. Althouse, and C.-I. Chang, "A quantitative and comparative analysis of linear and nonlinear spectral mixture models using radial basis function neural networks," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 39, no. 10, pp. 2314–2318, 2001.

[52] J. Plaza and A. Plaza, "Spectral mixture analysis of hyperspectral scenes using intelligently selected training samples," *IEEE Geoscience and Remote Sensing Letters*, vol. 7, no. 2, pp. 371–375, 2010.

[53] R. Guo, W. Wang, and H. Qi, "Hyperspectral image unmixing using autoencoder cascade," in *Proc. 7th Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing (WHISPERS)*, Tokyo, Japan, June 2015, pp. 1–4.

[54] 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.

[55] Y. Su, J. Li, A. Plaza, A. Marinoni, P. Gamba, and S. Chakravortty, "DAEN: Deep autoencoder networks for hyperspectral unmixing," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 57, no. 7, pp. 4309–4321, 2019.

[56] M. M. Sahoo, A. Porwal, A. Karnieli *et al.*, "Deep-learning-based latent space encoding for spectral unmixing of geological materials," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 183, pp. 307–320, 2022.

[57] B. Palsson, M. O. Ulfarsson, and J. R. Sveinsson, "Convolutional autoencoder for spectral-spatial hyperspectral unmixing," *IEEE Transactions on Geoscience and Remote Sensing*, pp. 1–15, 2020.

[58] Z. Hua, X. Li, J. Jiang, and L. Zhao, "Gated autoencoder network for spectral–spatial hyperspectral unmixing," *Remote Sensing*, vol. 13, no. 16, p. 3147, 2021.

[59] Y. Qu and H. Qi, "uDAS: An untied denoising autoencoder with sparsity for spectral unmixing," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 57, no. 3, pp. 1698–1712, March 2019.

[60] M. Zhao, M. Wang, J. Chen, and S. Rahardja, "Hyperspectral unmixing for additive nonlinear models with a 3-D-CNN autoencoder network," *IEEE Transactions on Geoscience and Remote Sensing*, 2021.

[61] K. T. Shahid and I. D. Schizas, "Unsupervised hyperspectral unmixing via nonlinear autoencoders," *IEEE Transactions on Geoscience and Remote Sensing*, 2021.

[62] H. Li, R. A. Borsoi, T. Imbiriba, P. Closas, J. C. Bermudez, and D. Erdoğmuş, "Model-based deep autoencoder networks for nonlinear hyperspectral unmixing," *IEEE Geoscience and Remote Sensing Letters*, vol. 19, pp. 1–5, 2021.

[63] R. A. Borsoi, T. Imbiriba, J. C. M. Bermudez, and C. Richard, "Deep generative models for library augmentation in multiple endmember spectral mixture analysis," *IEEE Geoscience and Remote Sensing Letters*, 2020.

[64] S. Shi, M. Zhao, L. Zhang, Y. Altmann, and J. Chen, "Probabilistic generative model for hyperspectral unmixing accounting for endmember variability," *IEEE Transactions on Geoscience and Remote Sensing*, 2021.

[65] T. Uezato, R. J. Murphy, A. Melkumyan, and A. Chlingaryan, "A novel spectral unmixing method incorporating spectral variability within endmember classes," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 54, no. 5, pp. 2812–2831, 2016.

[66] B. Koirala, Z. Zahiri, A. Lamberti, and P. Scheunders, "Robust supervised method for nonlinear spectral unmixing accounting for endmember variability," *IEEE Transactions on Geoscience and Remote Sensing*, 2020, doi: 10.1109/TGRS.2020.3031012.

[67] Q. Jin, Y. Ma, X. Mei, and J. Ma, "Tanet: An unsupervised two-stream autoencoder network for hyperspectral unmixing," *IEEE Transactions on Geoscience and Remote Sensing*, 2021.

[68] 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 Transactions on Neural Networks and Learning Systems*, 2021.

[69] H.-C. Li, X.-R. Feng, D.-H. Zhai, Q. Du, and A. Plaza, "Self-supervised robust deep matrix factorization for hyperspectral unmixing," *IEEE Transactions on Geoscience and Remote Sensing*, 2021.

[70] L. Drumetz, M. Dalla Mura, G. Tochon, and R. Fablet, "Learning endmember dynamics in multitemporal hyperspectral data using a state-space model formulation," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 2483–2487.

[71] A. Min, Z. Guo, H. Li, and J. Peng, "JMnet: Joint metric neural network for hyperspectral unmixing," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 60, pp. 1–12, 2021.

[72] Q. Jin, Y. Ma, F. Fan, J. Huang, X. Mei, and J. Ma, "Adversarial autoencoder network for hyperspectral unmixing," *IEEE Transactions on Neural Networks and Learning Systems*, 2021.

[73] L. Gao, Z. Han, D. Hong, B. Zhang, and J. Chanussot, "CyCU-Net: Cycle-consistency unmixing network by learning cascaded autoencoders," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 60, pp. 1–14, 2021.

[74] B. Rasti, B. Koirala, P. Scheunders, and P. Ghamisi, "UnDIP: Hyperspectral unmixing using deep image prior," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 60, pp. 1–15, 2021.

[75] S. Särkkä, *Bayesian filtering and smoothing*. Cambridge University Press, 2013, vol. 3.

[76] K. P. Murphy, *Machine learning: a probabilistic perspective*. MIT press, 2012.

[77] R. A. Borsoi, T. Imbiriba, and J. C. M. Bermudez, "Super-resolution for hyperspectral and multispectral image fusion accounting for seasonal spectral variability," *IEEE Trans. Image Process.*, vol. 29, no. 1, pp. 116–127, 2020.

[78] A. Halimi, P. Honeine, and J. M. Bioucas-Dias, "Hyperspectral unmixing in presence of endmember variability, nonlinearity, or mismodeling effects," *IEEE Transactions on Image Processing*, vol. 25, no. 10, pp. 4565–4579, 2016.

[79] A. Halimi, J. Bioucas-Dias, N. Dobigeon, G. S. Buller, and S. McLaughlin, "Fast hyperspectral unmixing in presence of nonlinearity or mismodelling effects," *IEEE Trans. Computational Imaging*, vol. 3, no. 2, pp. 146–159, April 2017.

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

[81] S. Jacquemoud and S. L. Ustin, "Leaf optical properties: A state of the art," in *8th International Symposium of Physical Measurements & Signatures in Remote Sensing*. CNES, Aussois France, 2001, pp. 223–332.

[82] A. Halimi, N. Dobigeon, and J.-Y. Tourneret, "Unsupervised unmixing of hyperspectral images accounting for endmember variability," *IEEE Trans. Image Processing*, vol. 24, no. 12, pp. 4904–4917, Dec. 2015.

[83] O. Eches, J. A. Benediktsson, N. Dobigeon, and J.-Y. Tourneret, "Adaptive Markov random fields for joint unmixing and segmentation of hyperspectral images," *IEEE Transactions on Image Processing*, vol. 22, no. 1, pp. 5–16, 2012.

[84] P. Hennig, D. Stern, R. Herbrich, and T. Graepel, "Kernel topic models," in *Artificial intelligence and statistics*. PMLR, 2012, pp. 511–519.

[85] A. Srivastava and C. Sutton, "Autoencoding variational inference for topic models," *arXiv preprint arXiv:1703.01488*, 2017.

[86] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," in *Proc. 2nd International Conference on Learning Representations (ICLR)*, Y. Bengio and Y. LeCun, Eds., Banff, AB, Canada, April 14-16, 2014.

[87] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. International Conf. on Learning Representations (ICLR)*, 2015.

[88] R. G. Krishnan, U. Shalit, and D. Sontag, "Structured inference networks for nonlinear state space models," in *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*, 2017, pp. 2101–2109.

[89] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[90] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," in *Proceedings of the thirteenth international conference on artificial intelligence and statistics*. JMLR Workshop and Conference Proceedings, 2010, pp. 249–256.

[91] J. M. P. Nascimento and J. M. Bioucas-Dias, "Vertex Component Analysis: A fast algorithm to unmix hyperspectral data," *IEEE Trans. Geosci. Remote Sens.*, vol. 43, no. 4, pp. 898–910, April 2005.

[92] C. Doersch, "Tutorial on variational autoencoders," *arXiv preprint arXiv:1606.05908*, 2016.