IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received November 10, 2018, accepted December 25, 2018, date of publication January 3, 2019, date of current version January 29, 2019.

Digital Object Identifier 10.1109/ACCESS.2018.2890723

Robust Hierarchical Learning for Non-Negative Matrix Factorization With Outliers

YINAN LI[1,2], MENG SUN[2], HUGO VAN HAMME[3], (Senior Member, IEEE), XIONGWEI ZHANG[2], AND JIBIN YANG[2]

[1]School of Space Information, Space Engineering University, Beijing 101416, China
[2]Department of Intelligent Information Processing, Army Engineering University, Nanjing 210007, China
[3]Center for Processing Speech and Images, Electrical Engineering Department (ESAT), KU Leuven, 3001 Leuven, Belgium

Corresponding author: Meng Sun (sunmengccjs@gmail.com)

This work was supported by the Natural Science Foundation of Jiangsu Province for Excellent Young Scholars under Grant BK20180080.

ABSTRACT Desirable properties of extensions of non-negative matrix factorization (NMF) include robustness in the presence of noises and outliers, ease of implementation, the guarantee of convergence, operation in an automatic fashion that trades off the balance between data approximation and model simplicity well, and the capability to model the inherently sequential structure of time-series signals. The state-of-the-art methods typically have only a subset of these aforementioned properties and seldom simultaneously possess them all. In this paper, we propose a novel approach that provides all these desirable properties by extending the automatic relevance determination framework in NMF from Tan and Févotte. Starting from an objective function derived from the maximum a posterior estimation of a Bayesian model, we develop majorization–minimization algorithms that work effectively to determine the correct model order, regardless of the impact of noise and outliers. Subsequently, we give a rigorous convergence analysis of the proposed algorithms. Moreover, convolutive bases are also incorporated in the basic model so that it is able to capture the richness of temporal continuity. We perform experiments on both synthetic and real-world data sets to show the efficiency and robustness of our approach.

INDEX TERMS Non-negative matrix factorization, automatic relevance determination, majorization-minimization, convolutive bases.

I. INTRODUCTION

Discovering a parsimonious set of core ingredients that can effectively represent given observations is probably one of the most principle requirements in machine learning and data mining. As witnessed by the vast variety of real-world applications [1]–[3], non-negative matrix factorization (NMF) [4] has become a popular dimensionality reduction technique with the property to discover comprehensive and meaningful parts-based representations. By imposing the non-negativity constraint, NMF encourages only additive combination of repetitive ''parts'' to compose the ''whole'' dataset without leading to subtraction or diminishment of any components [3]. Given a non-negative matrix $\boldsymbol{V}$ of dimension $F \times N$, the core problem in NMF is to factorize $\boldsymbol{V}$ into the product of two non-negative matrices $\boldsymbol{W}$ and $\boldsymbol{H}$ with size $F \times K$ and $K \times N$ such that $\boldsymbol{V} \approx \boldsymbol{WH}$. $\boldsymbol{W}$ is a set of bases or atoms that can be interpreted as constructive building blocks of the observation and we can also refer to it as a dictionary, whilst $\boldsymbol{H}$ is a set of activations that indicates the contribution of each basis to the observation. Since the number of bases K is typically much smaller than either F or N, the data dimension is significantly reduced when $FK + KN \ll FN$.

In the factorization of NMF, the selection of K is vital, because a dictionary with too few bases will not be able to provide an accurate modeling of $\boldsymbol{V}$ whereas a dictionary with too many bases tends to bring the risk of overfitting and may describe undesired trivial structures [5]. Ideally, the exact model order, i.e., the exact number of latent components within the ground truth observation (or the ground truth rank within the observation), is available before decomposition (e.g. when training a dictionary for a synthetic dataset with a fixed number of components). In this case, one can expect efficient extraction of the latent components by manually setting the number of bases equal to the model order that is already known. However, in real-world applications, this information is typically not available beforehand and the number of bases has to be estimated, which formulates the first challenge.

2169-3536 © 2019 IEEE. Translations and content mining are permitted for academic research only.
Personal use is also permitted, but republication/redistribution requires IEEE permission.
See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

The second challenge arises when the observation contains noise and outliers. Previous studies have shown that NMF is sensitive to outliers in the data matrix [6]. The outliers with large errors, if not handled properly, will dominate the objective function and cause severe distortion in the learned bases [7]. Meanwhile, outliers deteriorate the performance of the pruning of trivial bases, which is important for the automatic estimation of the model order in the observation, thus it becomes much more challenging to determine the ground truth model order from the corrupted observations with outliers.

We will encounter a third challenge when the consecutive columns of $\boldsymbol{V}$ exhibit strong temporal dynamics, e.g., in speech processing and the analysis of time series signals, where we have a necessity to model the inherently sequential structure. Indeed, in the setting of basic NMF, each observation is treated independently [8]. The negligence of temporal correlation has certainly limited the practical application of the basic NMF [9].

Many efforts have been devoted to address each challenge separately and a vast variety of algorithms have been explored. However, to the best of our knowledge, no algorithm exists for simultaneously addressing all the three challenges. Inspired by the recent work from Tan and Févotte [10], [11], we propose robust hierarchical learning for non-negative matrix factorization (RHL-NMF) which jointly considers all the aforementioned challenges. The optimization of RHL-NMF is based on a surrogate auxiliary function and is solved by the majorization-minimization strategy which yields multiplicative updating rules. The solution is a good compromise between efficiency and the ease of implementation. The merits of RHL-NMF are summarized as follows:

1) RHL-NMF is not only robust to dense noise such as Gaussian noise, but can also successfully recover the ground truth low-rank subspace when corrupted by outliers with large errors.
2) When the number of bases is larger than the oracle model order, RHL-NMF is capable to automatically shape some of the bases to the ground truth low-rank features while driving the remaining trivial bases to tiny values close to zero. Coupled with pruning strategies, we can realize automatic model order selection and obtain a parsimonious representation of the data.
3) RHL-NMF provides efficient and elegant updating rules that are compatible with standard NMF algorithms and we can also prove that the objective function of RHL-NMF monotonically decreases until it converges.
4) RHL-NMF yields a hierarchical representation by incorporating a set of extra parameters to indicate the contribution of each basis in representing the observation. The yielded pyramid structure allows us to evaluate the importance of each basis in a more intuitive fashion to make the pruning of the trivial bases more straightforward.
5) A convolutive extension is proposed and is combined with the above RHL-NMF framework to develop a powerful technique that is able to discover the temporal dependency within the observation, as well as to maintain the merits mentioned above.

The rest of the paper is organized as follows. In Section II, we first review previous publications that address each of the above three challenges in NMF. Subsequently, we describe the hierarchical Bayesian modeling framework in Section III. We present algorithms to solve RHL-NMF and analyze their convergence in Section IV. Section V provides a convolutive extension to RHL-NMF. We present the experimental results in Section VI and conclude the paper in Section VII.

II. REVIEW OF STATE-OF-THE-ART NMF METHODS

Among all the efforts that have been devoted to address different challenges that exist in NMF, we mainly mention three topics that have attracted much attention of researchers in this field.

A. NON-NEGATIVE LOW-RANK MODELING AND AUTOMATIC MODEL ORDER DETERMINATION

As one of the dimension reduction techniques, NMF should prefer a simple description rather than a complicated one when modeling observations. The optimal choice of K should be just equal to the number of latent components within the given observation $\boldsymbol{V}$ (i.e. the rank of the noise free matrix underlying $\boldsymbol{V}$) and we should strive to find the smallest set (i.e., the most parsimonious set) of bases that are ''just enough'' to capture the richness of the observation.

Several approaches including the most intuitive group sparsity [12], the computationally expensive Markov chain Monte Carlo (MCMC) [13], the greedy deflation method [14], automatic relevance determination (ARD) [15], [16], entropy regularization [17] and variational Bayesian based methods which calculate the marginal likelihood [1] have been investigated to seek such parsimonious representations. Despite the remarkable progress reported in the literature, state-of-the-art methods still suffer from critical disadvantages and limitations when dealing with real world problems. The pursuit for the ground truth non-negative model order is quite challenging because observations may deviate from the assumptions taken to simplify the model. Actually, previous studies have shown that these approaches are sensitive to noises and outliers in the observation; even a few data points with large corruptions may lead to complete failure [18]. Hence, robustness to noises and outliers is crucial in terms of automatic model order determination, and it is closely related to robust versions of non-negative matrix factorization which we will mainly discuss in the sequel of this paper.

B. ROBUST NON-NEGATIVE MATRIX FACTORIZATION

To reduce the effect of noises and outliers, some robustness strategies have been investigated and we will refer to them as robust non-negative matrix factorization (RNMF). The effort

to pursue such a RNMF can be roughly divided into two categories.

The first efficient approach is to use some robust objective function to modify the loss used in the standard NMF such as the squared Euclidean distance (SED) and Kullback-Leibler divergence (KLD) [19]–[21]. In [19], the cost function is replaced by the $\ell_{2,1}$-norm loss to prevent the outliers from dominating the objective function.

The second group of techniques for RNMF is to incorporate an extra additive term to accommodate outliers [22]–[26]. In this setting, the observation is reformulated as $\boldsymbol{V} \approx \boldsymbol{WH}+\boldsymbol{S}$. The optimization is hereby describe as,

$$\begin{aligned} &\underset{\boldsymbol{W},\boldsymbol{H},\boldsymbol{S}}{arg\,\min}\ O\left(\boldsymbol{W},\boldsymbol{H},\boldsymbol{S}\right) \\ &\quad = \underset{\boldsymbol{W},\boldsymbol{H},\boldsymbol{S}}{\arg\min}\, D\left(\boldsymbol{V}|\boldsymbol{WH}+\boldsymbol{S}\right)+R\left(\boldsymbol{S}\right) \\ &\qquad s.t.\quad \boldsymbol{W}\in\mathbb{R}_{\geq 0}^{F\times K},\quad \boldsymbol{H}\in\mathbb{R}_{\geq 0}^{K\times N},\ \boldsymbol{S}\in\mathbb{R}^{F\times N} \end{aligned} \tag{1}$$

where $D\left(\cdot|\cdot\right)$ is a measure of dissimilarity and $\boldsymbol{S}$ is the extra term accounting for the outliers. Most elements in $\boldsymbol{S}$ are zeros with a few outliers. Some previous studies have enforced the outliers to be non-negative for particular applications such as hyper-spectral unmixing [22], speech separation [26], speech enhancement [24], etc. Without loss of generality, the outliers here can include both positive and negative elements. Note that since the elements in $\boldsymbol{S}$ can be negative and arbitrarily large, we cannot guarantee the elements of the approximation $\boldsymbol{WH}+\boldsymbol{S}$ to be non-negative. The penalty term $R\left(\boldsymbol{S}\right)$ is used to promote the sparsity of $\boldsymbol{S}$. This type of RNMF originates from the recent development of convex optimization which makes full use of the relationship between sparsity and low-rank representation, and can be regarded as the non-negative counterpart of robust principle component analysis (RPCA) [27].

Despite the different formulations of RNMF, existing methods seldomly possess the property of automatic model order selection or rank estimation, to the best of our knowledge. Typically, the number of bases, K, is empirically chosen as a small integer to benefit from the low-rank approximation and all these bases are regarded equal in terms of data fitting, as was explored in [22], [23], and [25]. The limitation is obvious when the ground truth model order is larger than K where under-fitting happens. On the other hand, a manually selected K, which is larger than the ground truth model order, will bring the risk of splitting the ground truth bases and will yield trivial bases that have little contribution to model the input data. With the presence of noises, extra bases tend to represent the meaningless noises as well.

Because the statistical property can be greatly corrupted by outliers, it is challenging to estimate the ground truth model order of NMF with outliers in the input data. To the best of our knowledge, practical techniques have not been studied to conduct such model order selection with the disturbing outliers.

C. CONVOLUTIVE NON-NEGATIVE BASES LEARNING

A shortcoming of the basic NMF is that each column of $\boldsymbol{V}$ is treated independently, and it will not make a difference if we rearrange these columns. However, when processing time series, such as audio and speech, temporal dynamics are key properties and the consecutive columns of $\boldsymbol{V}$ (i.e., the adjacent frames of a spectrogram) form a sequence with evolving dynamics.

Existing dynamic extensions of NMF can be divided into two distinct categories. The first category imposes the temporal continuity on the activation matrix $\boldsymbol{H}$ [28]–[32]. These methods are flexible and easily combined with well-known dynamic models in signal processing such as discrete state hidden Markov models (HMM), vector autoregressive (VAR) models, smooth strategies, and so on. The second category is called convolutive NMF (CNMF) which formulates the temporal dependencies by incorporating an extra time dimension $\tau = 0,\ldots,T-1$ to the bases of $\boldsymbol{W}$ [23], [33]–[36]. The optimization problem of CNMF is thus reformulated as,

$$\begin{aligned} &\underset{\boldsymbol{W},\boldsymbol{H}}{arg\,\min}\, D\left(\boldsymbol{V}|\sum_{t=0}^{T-1}\boldsymbol{W}\left(t\right)\overset{t\rightarrow}{\boldsymbol{H}}\right) \\ &s.t.\ \boldsymbol{W}\left(t\right)\in\mathbb{R}_{\geq 0}^{F\times K}\quad \forall t,\ \boldsymbol{H}\in\mathbb{R}_{\geq 0}^{K\times N} \end{aligned} \tag{2}$$

where the bases matrix is in fact a tensor $\boldsymbol{W}\in\mathbb{R}_{\geq 0}^{F\times N\times T}$ where each $\boldsymbol{W}\left(t\right)$ represents a short time slice of the spectrogram. $\overset{t\rightarrow}{\boldsymbol{H}}$ shifts t columns to the right.

In this paper, we focus on the second category (i.e. CNMF) and further extend the scope of our previous study in [23] and [37]. We formulate a convolutive version of RHL-NMF to handle continuous dynamics in time series signals, as well as to conduct automatic model order determination robustly.

III. PROBLEM FORMULATION

In order to simultaneously handle the three aforementioned challenges, we employ a hierarchical Bayesian framework where all the observed and unknown variables are treated as statistical variables.

A. STATISTICAL ASSUMPTIONS

To begin with, we summarize some notations that are frequently used in the rest of this paper in Table 1.

Inspired by the ARD-NMF from Tan and Févotte [10], we assume that the elements of each basis and its corresponding activations are both generated from an exponential distribution with a common parameter, that is,

$$p\left(\boldsymbol{W}|\boldsymbol{\lambda}\right)=\prod_{f=1}^{F}\prod_{k=1}^{K}\mathcal{E}\left(W_{fk}\mid\lambda_k\right) \tag{3}$$

$$p\left(\boldsymbol{H}|\boldsymbol{\lambda}\right)=\prod_{k=1}^{K}\prod_{n=1}^{N}\mathcal{E}\left(H_{kn}\mid\lambda_k\right) \tag{4}$$

$$\mathcal{E}\left(x|\lambda\right)=\frac{1}{\lambda}\exp\left(-\frac{x}{\lambda}\right) \tag{5}$$

TABLE 1. Notations of variables and symbols used in this paper.

Notations	Description
f	Frequency index/ Row index
k	Basis index
n	Column index
t	Time index
V	A scalar value
$\boldsymbol{W}$	A matrix of random variables
W_{fk} or $[\boldsymbol{W}]_{fk}$	The element of the f-th row and the k-th column in matrix $\boldsymbol{W}$
$\boldsymbol{h_n} = \left[H_{1n}, \cdots, H_{kn}\right]^{\mathrm{T}}$	A column vector which is the n-th column of $\boldsymbol{H}$
h_i	The i-th element in vector $\boldsymbol{h}$
$\boldsymbol{H} = [\boldsymbol{h}_1, \cdots, \boldsymbol{h_n}]$	A matrix consisting of column vectors

where x is the random variable following an exponential distribution $\mathcal{E}(x|\lambda)$, and $x \geq 0, \lambda \geq 0$. $\boldsymbol{\lambda} = (\lambda_1, \lambda_2, \cdots, \lambda_K) \in \mathbb{R}^K_{\geq 0}$ refers to a group of relevance weights which is a set of non-negative parameters that associate the columns of $\boldsymbol{W}$ and their corresponding rows of $\boldsymbol{H}$. A relative large λ_k indicates great contribution of the k-th component to model the data, whilst a small one implies a weak relevance. As shown later, relevance weights of the irrelevant components to the dataset will be driven to tiny values close to zero, which eventually produces a parsimonious approximation of the observation.

In addition to (3) and (4), we impose the inverse-Gamma priors on the relevance weights,

$$\begin{aligned} p(\boldsymbol{\lambda}|a,b) &= \prod_{k=1}^{K} p(\lambda_k|a,b) \\ &= \prod_{k=1}^{K} \frac{b^a}{\Gamma(a)} \lambda_k^{-(a+1)} \exp\left(-\frac{b}{\lambda_k}\right) \end{aligned} \tag{6}$$

where non-negative parameters a and b are deterministic for modeling shape and scale, respectively. We assume that all the relevance weights $\lambda_k, \forall k$ share the same a and b.

Outliers are modeled by a sparse matrix $\boldsymbol{S}$. We utilize a double-exponential distribution for each element in $\boldsymbol{S}$ to represent sparse outliers. Therefore,

$$p(\boldsymbol{S}|\eta) = \prod_{f=1}^{F}\prod_{n=1}^{N} \mathcal{DE}\left(S_{fn} \mid \eta\right) \tag{7}$$

$$\mathcal{DE}(s|\eta) = \frac{1}{2\eta} \exp\left(-\frac{|s|}{\eta}\right) \tag{8}$$

where $s \in \mathbb{R}, \eta > 0$.

Likewise, we assign another level of hierarchy to the prior and impose the inverse-Gamma distribution to the hyper-parameter η which dominates the sparsity in $\boldsymbol{S}$.

$$p(\eta|c,d) = \frac{d^c}{\Gamma(c)} \eta^{-(c+1)} \exp\left(-\frac{d}{\eta}\right) \tag{9}$$

where c and d are the shape and scale parameter, respectively.

B. THE OBJECTIVE FUNCTION

Divergence metrics such as Squared Euclidean Distance (SED) and KLD are highly relevant to the family of Tweedie distributions as discussed in [38]. They can be regarded as pseudo-likelihoods, and can be derived by maximizing the likelihood of a generative model. If we assume that the noise is additive, independent and identically distributed Gaussian, we can get the following equation,

$$\begin{aligned} &-\log p(\boldsymbol{V}|\boldsymbol{W},\boldsymbol{H},\boldsymbol{S},\kappa) \\ &\qquad = \frac{\kappa}{2}\sum_{f=1}^{F}\sum_{n=1}^{N}\left([\boldsymbol{V}-\boldsymbol{W}\boldsymbol{H}-\boldsymbol{S}]_{fn}\right)^2 + cst \end{aligned} \tag{10}$$

where cst is a constant and κ is the noise intensity. Given the Gaussian noise assumption in (10), κ is determined by the variance of the Gaussian distribution. To determine the value of κ, a conventional strategy is to impose a prior $p(\kappa)$ and to optimize the objective function over κ. However, it is also reasonable to fix the noise intensity according to some prior knowledge on the variance. Like in [10], for simplicity, we take the second strategy in this paper. Note that there is no guarantee that elements in the approximation $\boldsymbol{WH} + \boldsymbol{S}$ are non-negative, thus, we are not going to discuss models with general divergence as in [39], but only concentrate on the SED here.

Under this condition, the prior $p(\kappa)$ is no longer an unknown parameter and the joint distribution can be expressed in a generative form where each factor is a prior or a conditional distribution, and hence

$$\begin{aligned} &p(\boldsymbol{V},\boldsymbol{W},\boldsymbol{H},\boldsymbol{S},\boldsymbol{\lambda},\eta) \\ &\quad = p(\boldsymbol{V}|\boldsymbol{W},\boldsymbol{H},\boldsymbol{S})\,p(\boldsymbol{W}|\boldsymbol{\lambda})\,p(\boldsymbol{H}|\boldsymbol{\lambda})\,p(\boldsymbol{S}|\eta)\,p(\boldsymbol{\lambda})\,p(\eta) \end{aligned} \tag{11}$$

By calculating the negative log-likelihood of the posterior distribution, we obtain an objective function. Minimizing this objective function yields the maximum a posterior (MAP) estimation of the joint distribution.

$$\begin{aligned} &O(\boldsymbol{W},\boldsymbol{H},\boldsymbol{S},\boldsymbol{\lambda},\eta) \\ &\quad = -\log p(\boldsymbol{W},\boldsymbol{H},\boldsymbol{S},\boldsymbol{\lambda},\eta \mid \boldsymbol{V}) \\ &\quad = \frac{\kappa}{2}\sum_{f=1}^{F}\sum_{n=1}^{N}\left([\boldsymbol{V}-\boldsymbol{W}\boldsymbol{H}-\boldsymbol{S}]_{fn}\right)^2 + U\log\lambda_k \\ &\qquad + \sum_{k=1}^{K}\frac{1}{\lambda_k}\left(\sum_{f=1}^{F}W_{fk} + \sum_{n=1}^{N}H_{kn} + b\right) + V\log\eta \\ &\qquad + \sum_{f=1}^{F}\sum_{n=1}^{N}\frac{1}{\eta}\left(|S_{fn}| + d\right) + cst \end{aligned} \tag{12}$$

where $U = F + N + a + 1$ and $V = F \times N + c + 1$. $\lambda_k, \forall k$ and η are strictly positive parameters.

The difference between our method and ARD-NMF is twofold. Firstly, there is no outlier model in ARD-NMF while there is one in our method. Secondly, we will also incorporate the temporal dynamics to our model by incorporating an extra time dimension, which we will present in Section V.

IV. ALGORITHM

In this section, we present the algorithm for minimizing the objective function of Section III by block coordinate

descent over $\boldsymbol{W}$, $\boldsymbol{H}$ and $\boldsymbol{S}$ alternatively. The updating scheme employed is Majorization-Minimization (MM) which yields multiplicative updates.

A. MAJORIZATION-MINIMIZATION ALGORITHMS

A key step for majorization-minimization algorithms is the construction of an auxiliary function. One choice of the auxiliary function can be defined as follows,

Definition 1: $G\left(\boldsymbol{h}|\tilde{\boldsymbol{h}}\right)$ is defined as an auxiliary function for $C(\boldsymbol{h})$ if and only if

$$G\left(\boldsymbol{h}|\tilde{\boldsymbol{h}}\right) \geq C(\boldsymbol{h}) \quad \textit{and } G(\boldsymbol{h}|\boldsymbol{h}) = C(\boldsymbol{h}) \tag{13}$$

It implies that the auxiliary function $G\left(\boldsymbol{h}|\tilde{\boldsymbol{h}}\right)$ is the upper bound of $C(\boldsymbol{h})$ and it is tight when $\boldsymbol{h} = \tilde{\boldsymbol{h}}$. With $\tilde{\boldsymbol{h}}$ fixed, the optimization of $G\left(\boldsymbol{h}|\tilde{\boldsymbol{h}}\right)$ over $\boldsymbol{h}$ is much more straightforward than the optimization of $C(\boldsymbol{h})$. We can thus derive the non-increasing updating rule via the optimization $\boldsymbol{h}^{(i+1)} = \arg\min_{\boldsymbol{h}} G\left(\boldsymbol{h}|\boldsymbol{h}^{(i)}\right)$.

Given $\boldsymbol{S}$, due to the symmetry of the objective function by transposition, i.e., the roles of $\boldsymbol{W}$ and $\boldsymbol{H}$ are equal because $\boldsymbol{V}-\boldsymbol{S} \approx \boldsymbol{W}\boldsymbol{H}$ is equivalent to $(\boldsymbol{V}-\boldsymbol{S})^{\mathrm{T}} \approx \boldsymbol{H}^{\mathrm{T}}\boldsymbol{W}^{\mathrm{T}}$, the update of $\boldsymbol{W}$ given $\boldsymbol{H}$ and the update of $\boldsymbol{H}$ given $\boldsymbol{W}$ are essentially the same. Thus, we will only present the optimization of $\boldsymbol{H}$ with $\boldsymbol{W}$ and $\boldsymbol{S}$ fixed without loss of generality.

B. UPDATING THE LOW-RANK APPROXIMATION

The low-rank approximation of the data matrix $\boldsymbol{V}$ is expressed by $\boldsymbol{W}\boldsymbol{H}$. According to (12), the objective function which aims to optimize $\boldsymbol{H}$ given $\boldsymbol{W}$ and $\boldsymbol{S}$ are treated as the superposition of a data fitting term $C(\boldsymbol{H})$ and a penalty term $R(\boldsymbol{H})$, leading to,

$$\begin{aligned}
&\arg\min_{\boldsymbol{H}} O(\boldsymbol{H}) \\
&= \arg\min_{\boldsymbol{H}} \left[C(\boldsymbol{H}) + R(\boldsymbol{H})\right] \\
&= \arg\min_{\boldsymbol{H}} \left[\frac{\kappa}{2}\|\boldsymbol{V}-\boldsymbol{W}\boldsymbol{H}-\boldsymbol{S}\|_2^2 + R(\boldsymbol{H})\right] \\
&= \arg\min_{\boldsymbol{H}} \left[\frac{\kappa}{2}\|\boldsymbol{V}-\boldsymbol{W}\boldsymbol{H}-\boldsymbol{S}\|_2^2 + \sum_{k=1}^{K}\frac{1}{2\lambda_k}\sum_{n=1}^{N}H_{kn}\right]
\end{aligned} \tag{14}$$

where the penalty term is used to choose a solution with parsimonious representations and conduct automatic model order selection.

To construct an auxiliary function to derive our algorithms, we first establish the following Lemmas.

Lemma 1: Suppose matrix $\boldsymbol{A}$ is positive semi-definite and is decomposed in positive and negative components as $\boldsymbol{A} = \boldsymbol{A}^{+} - \boldsymbol{A}^{-}$, where

$$\begin{aligned}
A_{ij}^{+} &= \begin{cases} A_{ij} & \text{if } A_{ij} > 0 \\ 0 & \text{otherwise} \end{cases} \\
A_{ij}^{-} &= \begin{cases} |A_{ij}| & \text{if } A_{ij} < 0 \\ 0 & \text{otherwise} \end{cases}
\end{aligned} \tag{15}$$

then, $G\left(\boldsymbol{h}|\tilde{\boldsymbol{h}}\right)$ in (16) is an auxiliary function for the objective function $C(\boldsymbol{h})$ expressed in (17).

$$\begin{aligned}
G\left(\boldsymbol{h}|\tilde{\boldsymbol{h}}\right) = \kappa \Bigg\{ &\frac{1}{2}\sum_{i=1}^{K}\frac{\left(\boldsymbol{A}^{+}\tilde{\boldsymbol{h}}\right)_i}{\tilde{h}_i}h_i^2 + \sum_{i=1}^{K}e_i h_i \\
&-\frac{1}{2}\sum_{i=1}^{K}\sum_{j=1}^{K}A_{ij}^{-}\tilde{h}_i\tilde{h}_j\left(1+\log\frac{h_i h_j}{\tilde{h}_i\tilde{h}_j}\right)\Bigg\}
\end{aligned} \tag{16}$$

$$\begin{aligned}
C(\boldsymbol{h}) = \kappa \Bigg\{ &\frac{1}{2}\sum_{i=1}^{K}\sum_{j=1}^{K}A_{ij}^{+}h_i h_j + \sum_{i=1}^{K}e_i h_i \\
&-\frac{1}{2}\sum_{i=1}^{K}\sum_{j=1}^{K}A_{ij}^{-}h_i h_j\Bigg\}
\end{aligned} \tag{17}$$

The above lemma can be shown and proved using quadratic programming, where the optimization is confined to an axis-aligned region in the nonnegative orthant, [40], [41].

Lemma 2: The function $f(x) = \frac{1}{2}\left(x^2-1\right) - (x-1)$, $x \in \mathbb{R}$ is non-negative, and its minimum value, 0, is reached when $x = 1$.

This result can be easily obtained by calculating the derivation. We can find that $f(x)$ is monotonous decreasing when $x < 1$ and monotonous increasing when $x > 1$. The minimum 0 is reached when $x = 1$.

As mentioned above, we split the objective function into a data fitting term and a penalty term where the data fitting term can be rewritten as follows,

$$\begin{aligned}
&\arg\min_{\boldsymbol{H}} C(\boldsymbol{H}) \\
&= \arg\min_{\boldsymbol{H}} \sum_{n=1}^{N} C(\boldsymbol{h_n}) \\
&= \arg\min_{\boldsymbol{H}} \sum_{n=1}^{N}\frac{\kappa}{2}\|\boldsymbol{v_n}-\boldsymbol{W}\boldsymbol{h_n}-\boldsymbol{s_n}\|_2^2 \\
&= \arg\min_{\boldsymbol{H}} \sum_{n=1}^{N}\kappa\left[\frac{1}{2}tr\left(\boldsymbol{h_n}^{\mathrm{T}}\boldsymbol{W}^{\mathrm{T}}\boldsymbol{W}\boldsymbol{h_n}\right) + tr\left((\boldsymbol{s_n}-\boldsymbol{v_n})^{\mathrm{T}}\boldsymbol{W}\boldsymbol{h_n}\right)\right]
\end{aligned} \tag{18}$$

where $\boldsymbol{H} = [\boldsymbol{h_1}, \cdots, \boldsymbol{h_n}]$ and $\boldsymbol{h_n} = [H_{1n}, \cdots, H_{kn}]^{\mathrm{T}}$. Analogously, $\boldsymbol{V} = [\boldsymbol{v_1}, \cdots, \boldsymbol{v_n}]$ and $\boldsymbol{S} = [\boldsymbol{s_1}, \cdots, \boldsymbol{s_n}]$.

On one hand, by applying Lemma 1 and on account of (16), (17) and (18), an auxiliary function $G\left(\boldsymbol{H}|\tilde{\boldsymbol{H}}\right)$ for the data fitting term is constructed when we set $\boldsymbol{A}^{+} = \boldsymbol{W}^{\mathrm{T}}\boldsymbol{W}$, $\boldsymbol{A}^{-} = \boldsymbol{0}$ and $\boldsymbol{E} = (\boldsymbol{S}-\boldsymbol{V})^{\mathrm{T}}\boldsymbol{W}$.

On the other hand, by following Lemma 2 we can get the following result if we let $x = H_{kn}\big/\tilde{H}_{kn}$. The equality is reached when $x = 1$, i.e., $H_{kn} = \tilde{H}_{kn}$.

$$\frac{H_{kn}}{\tilde{H}_{kn}} - 1 \leq \frac{1}{2}\left[\left(\frac{H_{kn}}{\tilde{H}_{kn}}\right)^2 - 1\right] \tag{19}$$

which can be reformulated as

$$\frac{1}{\lambda_k}H_{kn} \leq \frac{1}{2\lambda_k}\tilde{H}_{kn}\left(\frac{H_{kn}}{\tilde{H}_{kn}}\right)^2 + \frac{\tilde{H}_{kn}}{2\lambda_k} \tag{20}$$

Thus, we can construct an auxiliary function $F\left(\boldsymbol{H}|\tilde{\boldsymbol{H}}\right)$ for $O(\boldsymbol{H})$ as follows,

$$F\left(\boldsymbol{H}|\tilde{\boldsymbol{H}}\right) = G\left(\boldsymbol{H}|\tilde{\boldsymbol{H}}\right) + \sum_{k=1}^{K}\sum_{n=1}^{N}\frac{1}{2\lambda_k}\tilde{H}_{kn}\left(\frac{H_{kn}}{\tilde{H}_{kn}}\right)^2 + \frac{\tilde{H}_{kn}}{2\lambda_k} \tag{21}$$

Suppose that $H_{k'n'}$ is one of the specific constituent elements in $\boldsymbol{H}$, calculating the partial derivative of $F\left(\boldsymbol{H}|\tilde{\boldsymbol{H}}\right)$ with respect to $H_{k'n'}$ and setting it to zero, we obtain a quadratic equation,

$$\begin{aligned}
&\frac{\partial F}{\partial H_{k'n'}} \\
&= \frac{\partial G}{\partial H_{k'n'}} + \frac{1}{\lambda_{k'}}\cdot\frac{H_{k'n'}}{\tilde{H}_{kn}} \\
&= \frac{\kappa\left[\boldsymbol{A}^{+}\tilde{\boldsymbol{H}}\right]_{k'n'}}{\tilde{H}_{k'n'}}H_{k'n'} + \kappa E_{k'n'} + \frac{1}{\lambda_{k'}}\cdot\frac{H_{k'n'}}{\tilde{H}_{k'n'}} - \frac{\kappa}{2} \\
&\quad\times\left(\sum_{i=1}^{K}A^{-}_{ij}\tilde{H}_{in'}\tilde{H}_{k'n'}\frac{1}{H_{k'n'}} + \sum_{j=1}^{K}A^{-}_{ij}\tilde{H}_{k'n'}\tilde{H}_{jn'}\frac{1}{H_{k'n'}}\right) = 0
\end{aligned} \tag{22}$$

Because $\boldsymbol{A}$ is a symmetrical matrix, we obtain a positive root from the quadratic equation,

$$\begin{aligned}
&H_{k'n'} \\
&= \kappa\tilde{H}_{k'n'}\frac{-E_{k'n'} + \sqrt{E^2_{k'n'} + 4\left(\left[\boldsymbol{A}^{+}\tilde{\boldsymbol{H}}\right]_{k'n'} + \frac{1}{\lambda_{k'}}\right)\left[\boldsymbol{A}^{-}\tilde{\boldsymbol{H}}\right]_{k'n'}}}{2\left(\kappa\left[\boldsymbol{A}^{+}\tilde{\boldsymbol{H}}\right]_{k'n'} + \frac{1}{\lambda_{k'}}\right)}
\end{aligned} \tag{23}$$

Substituting $\boldsymbol{E}$, $\boldsymbol{A}^{+}$ and $\boldsymbol{A}^{-}$ with $(\boldsymbol{S}-\boldsymbol{V})^{\mathrm{T}}\boldsymbol{W}$, $\boldsymbol{W}^{\mathrm{T}}\boldsymbol{W}$ and $\boldsymbol{0}$, respectively, we obtain the updating equation for $\boldsymbol{H}$,

$$\boldsymbol{H} \leftarrow \kappa\boldsymbol{H} \odot \frac{\left|\boldsymbol{W}^{\mathrm{T}}(\boldsymbol{S}-\boldsymbol{V})\right| - \left(\boldsymbol{W}^{\mathrm{T}}(\boldsymbol{S}-\boldsymbol{V})\right)}{2\kappa\left(\boldsymbol{W}^{\mathrm{T}}\boldsymbol{W}\boldsymbol{H}\right) + 2 \oslash \left(\boldsymbol{\lambda}^{\mathrm{T}}\boldsymbol{1}_{\boldsymbol{H}}\right)} \tag{24}$$

where $\boldsymbol{1}_{\boldsymbol{H}} \in \mathbb{R}^{1\times N}$ is an all-one vector and we use notations $\odot$ to denote element-wise multiplication, and $\oslash$ to denote the operation that a constant is divided by each element of a matrix, respectively. In addition, the notation for matrix division is also conducted in an element-wise fashion.

Likewise, we can derive the updating equation for $\boldsymbol{W}$ as:

$$\boldsymbol{W} \leftarrow \kappa\boldsymbol{W} \odot \frac{\left|(\boldsymbol{S}-\boldsymbol{V})\boldsymbol{H}^{\mathrm{T}}\right| - \left((\boldsymbol{S}-\boldsymbol{V})\boldsymbol{H}^{\mathrm{T}}\right)}{2\kappa\left(\boldsymbol{W}\boldsymbol{H}\boldsymbol{H}^{\mathrm{T}}\right) + 2 \oslash (\boldsymbol{1}_{\boldsymbol{W}}\boldsymbol{\lambda})} \tag{25}$$

where $\boldsymbol{1}_{\boldsymbol{W}} \in \mathbb{R}^{F\times 1}$ is once again used to denote an all-one vector.

C. UPDATING THE SPARSE MATRIX

The MAP estimate of the outliers associated with the double-exponential prior is identical to the convex optimization problem listed below,

$$\begin{aligned}
\arg\min_{\boldsymbol{S}} O(\boldsymbol{S}) = \arg\min_{\boldsymbol{S}} \sum_{f=1}^{F}\sum_{n=1}^{N}\Bigg\{&\frac{\kappa}{2}\left([\boldsymbol{V}-\boldsymbol{W}\boldsymbol{H}-\boldsymbol{S}]_{fn}\right)^2 \\
&+\frac{1}{\eta}\left|S_{fn}\right|\Bigg\}
\end{aligned} \tag{26}$$

This is a Least Absolute Shrinkage and Selection Operator (LASSO) problem which can be solved efficiently via the soft-threshold operator denoted as $\mathcal{T}_v(\cdot)$,

$$[\boldsymbol{S}]_{fn} \leftarrow \mathcal{T}_{1/(2\kappa\eta)}\left([\boldsymbol{V}-\boldsymbol{W}\boldsymbol{H}]_{fn}\right) \tag{27}$$

where $\mathcal{T}_v(x) = sign(x)\max(|x|-v, 0)$ and v is the threshold parameter.

D. UPDATING THE HYPER-PARAMETERS

We have considered the variables $\boldsymbol{W}$, $\boldsymbol{H}$ and $\boldsymbol{S}$. In the hierarchical Bayesian framework, the statistics of these variables are governed by hyper-parameters $\boldsymbol{\lambda}$ and η, $\eta > 0$. The updates for these hyper-parameters are easily derived if we calculate the roots of partial derivative of $O(\boldsymbol{W}, \boldsymbol{H}, \boldsymbol{S}, \boldsymbol{\lambda}, \eta)$ with respect to $\lambda_k, \forall k \in [1, \cdots, K]$ and η respectively. In summary, we have the following updating rule for λ_k,

$$\begin{aligned}
\lambda_k &= \frac{\sum_{f=1}^{F}[\boldsymbol{W}]_{fk} + \sum_{n=1}^{N}[\boldsymbol{H}]_{kn} + b}{U} \\
&= \frac{\sum_{f=1}^{F}[\boldsymbol{W}]_{fk} + \sum_{n=1}^{N}[\boldsymbol{H}]_{kn} + b}{F+N+a+1}
\end{aligned} \tag{28}$$

For η, we have,

$$\begin{aligned}
\eta &= \frac{\sum_{f=1}^{F}\sum_{n=1}^{N}\left(\left|[\boldsymbol{S}]_{fn}\right| + d\right)}{V} \\
&= \frac{\sum_{f=1}^{F}\sum_{n=1}^{N}\left(\left|[\boldsymbol{S}]_{fn}\right| + d\right)}{F\times N + c + 1}
\end{aligned} \tag{29}$$

From the objective function in (12), we can observe that the noise intensity κ actually works to make a trade-off between the data fitting term and the penalty term. Given a certain value of κ, the relative scale of the data fitting term with respect with the penalty term is fixed. Then, RHL-NMF iteratively updates $\boldsymbol{W}, \boldsymbol{H}, \boldsymbol{S}, \boldsymbol{\lambda}$ and η until the whole objective function converges.

E. CONVERGENCE ANALYSIS

In this subsection, we show the monotonous decreasing of the cost function in RHL-NMF.

Theorem 1: Given the noise intensity κ, the objective function of RHL-NMF monotonously decreases until the

algorithm converges by following the updating equations in (24), (25), (27), (28) and (29).

Proof: In the i-th iteration, we denote the objective function after calculating (24), (25), (27), (28) and (29) as $O_1^{(i)}$, $O_2^{(i)}$, $O_3^{(i)}$, $O_4^{(i)}$ and $O_5^{(i)}$, respectively.

Suppose that we use θ to denote either $\boldsymbol{W}$ or $\boldsymbol{H}$. Then, because (24) and (25) are essentially optimizing $O(\theta)$ via $\theta = \arg_\theta \min G\left(\theta|\hat{\theta}\right)$, we have:

$$O\left(\theta^{(i+1)}\right) \leq G\left(\theta^{(i+1)}|\theta^{(i)}\right) \leq G\left(\theta^{(i)}|\theta^{(i)}\right) = O\left(\theta^{(i)}\right) \tag{30}$$

That is, $O_5^{(i)} \geq O_1^{(i+1)} \geq O_2^{(i+1)}$.

In addition, the soft-threshold operator yields a global optimal solution which implies that $O_2^{(i+1)} \geq O_3^{(i+1)}$. The updating of $\boldsymbol{\lambda}$ and η provides the optimal solution under the current parameters setting, i.e., $\boldsymbol{W}^{(i+1)}$, $\boldsymbol{H}^{(i+1)}$ and $\boldsymbol{S}^{(i)}$, leading to $O_3^{(i+1)} \geq O_4^{(i+1)} \geq O_5^{(i+1)}$. Therefore, if we update the parameters alternatively, a decreasing sequence is obtained as follows,

$$\begin{aligned} O_1^1 \geq \cdots \geq O_5^{(i)} \geq O_1^{(i+1)} \geq O_2^{(i+1)} \geq O_3^{(i+1)} \\ \geq O_4^{(i+1)} \geq O_5^{(i+1)} \geq \cdots \end{aligned} \tag{31}$$

When the objective function stops decreasing, the algorithm converges, which completes the proof.

F. SIMPLIFICATION FOR PARAMETER TUNING AND COMPUTATION

By revisiting (26), (27) and (29), it is obvious that we are essentially using hyper-parameters c and d to control the regularization parameter of the ℓ_1-norm penalty term, which is not straightforward.

A more simple and straightforward alternative is to alternate c and d by introducing a hyper-parameter γ to control the sparse matrix directly, which is in line with the cases in [7], [18], and [22]. In this setting, the optimization problem in (26) is rewritten as,

$$\begin{aligned} &\arg\min_{\boldsymbol{S}} O(\boldsymbol{S}) \\ &= \arg\min_{\boldsymbol{S}} \sum_{f=1}^{F}\sum_{n=1}^{N}\left\{\frac{1}{2}\left([\boldsymbol{V}-\boldsymbol{W}\boldsymbol{H}-\boldsymbol{S}]_{fn}\right)^2 + \gamma\left|S_{fn}\right|\right\} \end{aligned} \tag{32}$$

And the update for the sparse matrix (27) is rewritten as:

$$[\boldsymbol{S}]_{fn} \leftarrow \mathcal{T}_{\gamma/(2\kappa)}\left([\boldsymbol{V}-\boldsymbol{W}\boldsymbol{H}]_{fn}\right) \tag{33}$$

The above alternative works essentially the same as the previous one. However, it will bring some convenience in terms of parameter tuning and computation. On one hand, tuning one hyper-parameter is much easier than tuning two, i.e., tuning γ versus tuning c and d. On the other hand, the algorithm is accelerated because the computation cost to update η is reduced.

V. COVOLUTIVE VERSION OF RHL-NMF

In this section, we extend the basic RHL-NMF by introducing convolutive bases, which is well-suited to model the temporal dynamics of time series. The simple basis vectors are extended to multidimensional bases that span a given number of temporal frames.

Supposing that T frames are used to represent a basis, then for the k-th basis, all T frames are assumed to be dominated by a common parameter λ_k.

$$p(\boldsymbol{W}|\lambda) = \prod_{t=0}^{T-1}\prod_{f=1}^{F}\prod_{k=1}^{K}\mathcal{E}\left([\boldsymbol{W}(t)]_{fk}|\lambda_k\right) \tag{34}$$

We follow the standard assumption that the noise is approximately white with zero mean and variance κ. Then, to maximize the likelihood of data fitting is equivalents to minimizing the following objective function,

$$\begin{aligned} &\underset{\boldsymbol{W},\boldsymbol{H}}{arg\min}\ \frac{\kappa}{2}\left\|\boldsymbol{V}-\left(\sum_{t=0}^{T-1}\boldsymbol{W}(t)\overrightarrow{\boldsymbol{H}}^{t}+\boldsymbol{S}\right)\right\|_F^2 \\ &s.t.\ \boldsymbol{W}(t)\in\mathbb{R}_{\geq 0}^{F\times K}\quad \forall t,\ \boldsymbol{H}\in\mathbb{R}_{\geq 0}^{K\times N},\ \boldsymbol{S}\in\mathbb{R}^{F\times N} \end{aligned} \tag{35}$$

where the notation $\overrightarrow{(\cdot)}$ denotes the shifted right operation with left zero padding while $\overleftarrow{(\cdot)}$ denotes the opposite operation later.

TABLE 2. Algorithom to solve RHL-NMF.

Input	$\boldsymbol{V}, a, b, \gamma, T, \kappa$
Initialize	$\boldsymbol{W}, \boldsymbol{H}, \boldsymbol{S}$
Repeat the following steps until it converges to a local minimum	
Step1:	$\boldsymbol{H} \leftarrow \frac{\kappa}{T}\sum_{t=0}^{T-1}\boldsymbol{H}\odot \frac{\left\lfloor \boldsymbol{W}(t)^{\mathrm{T}}\overleftarrow{(\boldsymbol{S}-\boldsymbol{V})}^{t}\right\rfloor - \left[\boldsymbol{W}(t)^{\mathrm{T}}\overleftarrow{(\boldsymbol{S}-\boldsymbol{V})}^{t}\right]}{2\left\{\kappa\boldsymbol{W}(t)^{\mathrm{T}}\cdot\left[\sum_{\tau=0}^{T-1}\boldsymbol{W}(\tau)\overrightarrow{\boldsymbol{H}}^{\tau}+\boldsymbol{S}\right]+\frac{1}{\lambda^T\mathbf{1}_{\boldsymbol{H}}}\right\}}$
Step2:	$\boldsymbol{W}(t) \leftarrow \kappa\boldsymbol{W}(t)\odot \frac{\left\{\left\lfloor(\boldsymbol{S}-\boldsymbol{V})\cdot\overrightarrow{\boldsymbol{H}}^{t\mathrm{T}}\right\rfloor - \left[(\boldsymbol{S}-\boldsymbol{V})\cdot\overrightarrow{\boldsymbol{H}}^{t\mathrm{T}}\right]\right\}}{2\left\{\kappa\left(\sum_{\tau=0}^{T-1}\boldsymbol{W}(\tau)\overrightarrow{\boldsymbol{H}}^{\tau}+\boldsymbol{S}\right)\cdot\overrightarrow{\boldsymbol{H}}^{t\mathrm{T}}+\frac{1}{\mathbf{1}_{\boldsymbol{W}}\lambda}\right\}}$
Step3:	$[\boldsymbol{S}]_{fn} \leftarrow \mathcal{T}_{\gamma/(2\kappa)}\left(\left[\boldsymbol{V}-\sum_{t=0}^{T-1}\boldsymbol{W}(t)\overrightarrow{\boldsymbol{H}}^{t}\right]_{fn}\right)$
Step4:	$\lambda_k \leftarrow \frac{\frac{1}{T}\sum_{f=1}^{F}\sum_{t=0}^{T-1}[\boldsymbol{W}(t)]_{fk}+\sum_{n=1}^{N}[\boldsymbol{H}]_{kn}+b}{F+N+a+1}$

The updating equations for the convolutive RHL-NMF (CRHL-NMF) were derived by following [23] and [37]. The step-by-step algorithm is presented in Table 2. Note that RHL-NMF is a special case of CRHL-NMF with $T = 1$. Besides, when we update $\boldsymbol{W}(t)$ and $\boldsymbol{H}$ for each t, the fact that each time slice shares the same $\boldsymbol{H}$ will eventually lead a biased estimation, with the update for $t = T-1$ dominating over others [33]. To eliminate this effect, we take the average of all the updates as shown in Step1 of Table 2.

VI. EXPERIMENTS

In this section, we provide experimental results for RHL-NMF and CRHL-NMF on both synthetically generated

and real-world datasets. The main motivation here is not to evaluate the performance under multiple different tasks, where a thorough comparison with state-of-the-art approaches is required. Instead, we would like to demonstrate that the proposed algorithms process a couple of desirable properties through intuitive experiments and show that the proposed algorithm can deal with all the three challenges mentioned above, at the same time. To the best of our knowledge, this is the first algorithm that can simultaneously address all these challenges. We first conduct experiments on intuitive synthetic datasets to evaluate the performance of RHL-NMF and CRHL-NMF and then evaluate them on real-world datasets.

A. MODEL ORDER SELECTION ON NOISY SWIMMER DATASET

The swimmer dataset is a synthetic dataset that contains 256 images with size 32×32 [42]. Each image represents a swimmer composed of an invariant torso and four limbs, where each limb can take one of four positions. Hence, the ground truth model order K for the swimmer dataset should be either $K = 16$ or $K = 17$. In the case that $K = 16$, the invariant torso is associated with all the 16 different limb positions, while in the case that $K = 17$, the torso is regarded as a separate component apart from the 16 different limb positions.

To construct a noisy version of the swimmer dataset, the background pixel values are set to 1 and the body pixel values are set to 10, and Poisson noise is imposed to the dataset by following the recipes in [10]. Then, we generate a sparse matrix $\boldsymbol{S} \in \mathbb{R}^{F\times N}$ whose non-zero elements are located uniformly randomly and are draw from a uniform distribution in the range of $(-20, 20)$. To have an intuitive understanding of the scale of outliers, the mean and standard deviation of the adjusted swimmer dataset in [10] without outliers are around 1.35 and 2.17, respectively. The total number for non-zero elements is set to $0.05 \times F \times N$ and are added to the original data. That is, each element in the dataset has a probability of 5% being an outlier. Finally, we calculate the absolute value of the data to make it nonnegative so that we can compare our algorithm with NMF-based ones.

The first 10 images in the noisy swimmer dataset are shown in Fig.1(a). The output results of our algorithm are shown in Fig.1(b) and Fig.1(c), respectively. Specifically, Fig.1(b) is the low-rank approximation, i.e. $\boldsymbol{WH}$ in Eq. (10) whilst Fig.1(c) is the sparse matrix that accommodates for outliers, i.e. $\boldsymbol{S}$ in Eq. (10).

To make a fair comparison with ℓ_1-ARD NMF in [10], we run both RHL-NMF and ARD NMF on the above noisy swimmer dataset and initialized them with $K = 32$, which is a much larger value than ground truth and can be regarded as an upper bound of the number of bases. The intensity of the Gaussian noise is fixed to 1, i.e., $\kappa = 1$. We follow the parameter tuning strategy in [10], where a thorough study of parameter tuning is presented. a is set to 10 and b is calculated

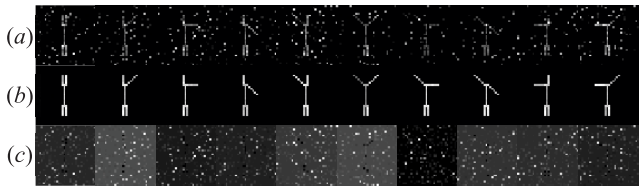


FIGURE 1. Experimental results for the noisy swimmer dataset. The first row (a) shows the original noisy images that are corrupted by noises and outliers. The second and the third rows ((b) and (c)) show the output results of our algorithm. Specifically, the row (b) lists the low-rank approximations (i.e. WH in Eq. (10)) whilst the row (c) lists the sparse matrices that accommodates for outliers (i.e. S in Eq. (10)). It is clearly seen from the figure that our algorithm successfully removes noises (c) from (a) by yielding (b).

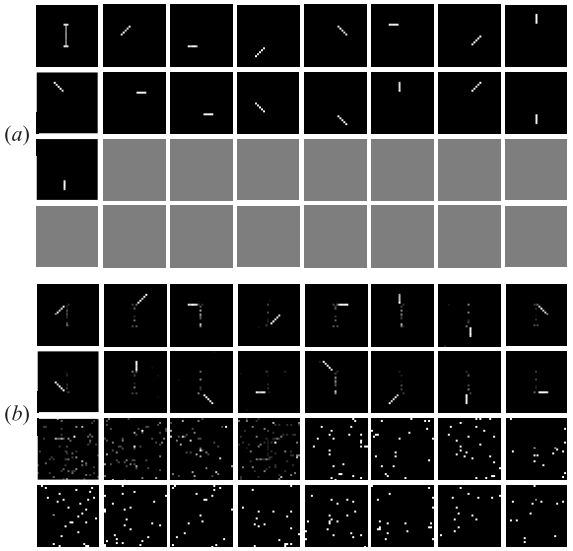


FIGURE 2. The learned components of RHL-NMF and ARD-NMF on the noisy swimmer dataset with outliers are presented in (*a*) and (*b*), respectively. The components are sorted in a decreasing order according to their corresponding relevance values from left to right, top to bottom. The model order of RHL-NMF turns out to be 17 where the invariant torso is recognized as a separate component, which is consistent with the ground truth.

according to Eq. (36),

$$b = \sqrt{\frac{(a-1)\,(a-2)}{F \times N \times K} \sum_{fn} \boldsymbol{V}_{fn}} \tag{36}$$

For the parameter γ which controls the weight of the sparsity penalty term, it is still an open problem how to select an appropriate value and has not been settled completely. In [22], an algorithm based on the method of moment is proposed. However, the authors also claim that the method is only a handy gross estimation that comes with no statistical guarantee. In this experiment, we set $\gamma = 1$ emperically.

The obtained results of RHL-NMF are demonstrated in Fig.1, Fig.2(a) and Fig.3(a), respectively. For comparison, the corresponding results of ARD-NMF [10] are shown in Fig.2(b) and Fig.3(b), respectively. In Fig.2, the components are sorted in a decreasing order according to their corresponding relevance values from left to right, top to bottom.

The model order of RHL-NMF turns out to be 17 where the invariant torso is recognized as a separate component. In Fig.3, the left column (i.e., Fig.3(a)) and the right column

(i.e., Fig.3(*b*)) are the results of RHL-NMF and ARD-NMF, respectively. The horizontal or lateral axis indicates the number of iteration while the vertical axis indicates the numeric value of relevance weights. In both Fig.3 (*a*) and Fig.3 (*b*), the top panel shows the evolution of the objective function as a function of iterations, whilst the middle panel shows the evolution of the data fitting term, and the bottom panel shows the evolving of the relevance values with iterations.

In Fig.1, we observe that the original clean images are successfully estimated and recovered (Fig.1(*b*)) from the corrupted ones (Fig.1(*a*)), and that the outliers are isolated by the sparse matrix $\boldsymbol{S}$ (Fig.1(*c*)). Because there is no consideration for the outliers in the framework of ARD-NMF, there is no improvement of the data quality in the result of ARD-NMF.

A further inspection of the output components and their corresponding learning process is provided in Fig.2 and Fig.3, respectively. Note that in both RHL-NMF and ARD-NMF, the contribution of each component in representing the observation can be quantified by its relevance value. Fig.2 visualizes the learned dictionary. The yielded components are sorted and displayed in a descending order according to their relevance values. We can see that the learned components of RHL-NMF are clean and in consistent with the data that is not corrupted by outliers.

By contrast, we observe that the components of ARD-NMF are affected by the outliers, especially by the torso. The residual of the outliers is visible in the background of each component, which makes the discovered components to deviate from the ground truth to some extent. Besides, the yielded results are not parsimonious because some extra bases are used to capture meaningless outliers.

Fig.3 illustrates the evolution of the learning process. The left column shows the results of RHL-NMF while the right column are the results of ARD-NMF. The monotonous decreasing of the objective function of RHL-NMF is consistent with our previous proof as shown in Fig.3 (*a*).

From these results in Fig.3 (*a*), we find that despite the interference of noise and outliers, the learned components are consistent with our knowledge about the ground truth order of the swimmer dataset. Here, it keeps in line with the previous claimed $K = 17$ components version where the invariant torso is recognized as a separate component. After an adequate number of iterations, these components are eventually reshaped to the torso and limbs of the swimmer. Moreover, it is easy to see that the relevance values fit well with the occurrence frequencies of their corresponding components. For instance, after 5000 iterations when all relevance values become steady, we find there is one relevance value which is much larger than the rest. That is because the invariant torso holds the highest frequency of appearance in the dataset.

By contrast, the 16 different positions of limbs have almost equal frequency of appearance and their relevance values are also similar, as shown in the plotted lines of the bottom panel of Fig.3 (*a*). Another result worth to mention is that the remaining 15 relevance values corresponding to the redundant bases are gradually driven to tiny values very close to zero, leading to a parsimonious representation after pruning [43]. Thus, the proposed algorithm is able to conduct model order selection on datasets with corruption of noise and outliers. For comparison, none of the relevance values of ARD-NMF are driven to zero due to the impact of outliers, as shown in Fig.3 (*b*).

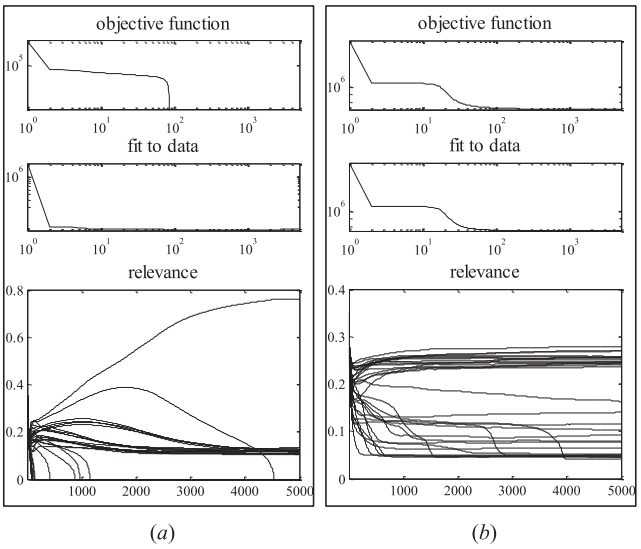


FIGURE 3. A comparison of the convergence of RHL-NMF (the left column (*a*)) and ARD-NMF (the right column (*b*)). In both (*a*) and (*b*), the top panel shows the evolution of the objective function as a function of iterations, whilst the middle panel shows the evolution of the goodness of data fitting, and the bottom panel shows the variation of the relevance values along with iterations. It can be seen from the curves of the bottom panel of (a) and (b) that a lot of relevance values in RHL-NMF become zero to yield parsimonious solutions, while ARD-NMF not.

B. MODEL ORDER SELECTION WITH A NOISY FREQUENCY-MODULATED SIGNAL

Temporal continuity is an important property when discovering patterns from time series signals. To evaluate the performance of the proposed covolutive RHL-NMF (CRHL-NMF) on capturing the temporal dependency of sequential signals, we conduct a group of intuitive experiments on the spectrogram of a synthetic noisy sine-like frequency-modulated signals, whose oracle number of latent components is easily observed.

In the spectrogram of the noisy frequency-modulated signal, there is a sine-like pattern which spans 16 frames, and the same pattern repeats again and again to constitute the whole signal. Subsequently, we add outliers to the spectrogram where each element has a 5% chance to be an outlier. The yielded spectrogram can be seen in Fig.4 (*a*). As we see from the figure, the ground truth model order is closely related to the selection of the pattern length. For example, the ground truth model order for the sine-like signal is 1 when the pattern length is set to $T = 16$ while the model order will turn out to be 2 if the pattern length is selected as $T = 8$.

We initialize CRHL-NMF with 5 convolutive bases and test CRHL-NMF under the conditions where $T = 8$ and $T = 16$, respectively. The experimental results are shown

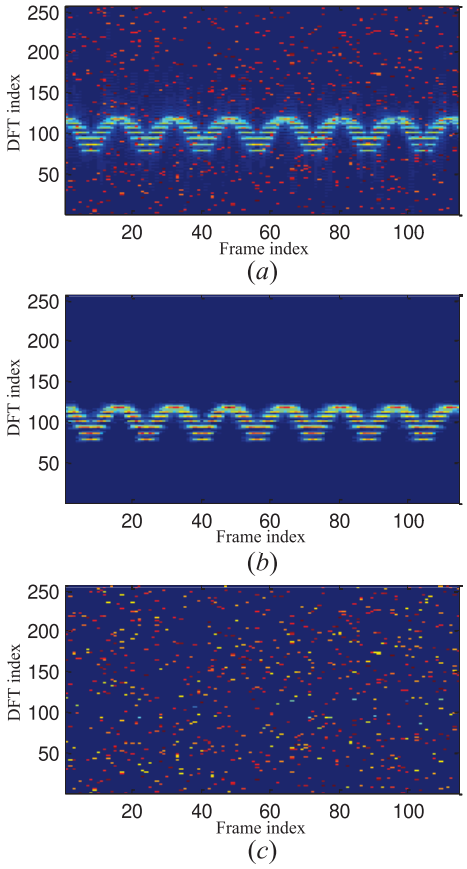


FIGURE 4. The top panel (*a*) shows the spectrogram of the noisy frequency-modulated signal. The middle panel (*b*) demonstrates the reconstructed low-rank component whilst the bottom panel (*c*) is the sparse matrix that accommodates for outliers.

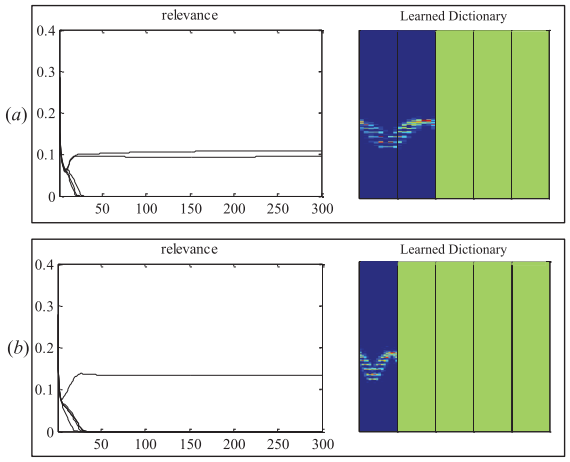


FIGURE 5. The top panel (*a*) shows the learned dictionary of which each basis with temporal span 8 (i.e. T=8) and the evolving of the relevance values during the learning process, whilst the bottom panel (*b*) presents those results for T=16. The blocks with blue background correspond to non-trivial bases, while the blocks with green background correspond to trivial bases with almost zero entries. In both cases, parsimonious solutions are obtained as expected, i.e. using as few as possible non-trivial bases to represent the data.

in Fig. 4 and Fig.5. In Fig.4, we can see that the low-rank structural signal, i.e., Fig.4 (*b*), is successfully recovered using the approximation $\sum_{t=0}^{T-1} \boldsymbol{W}(t) \overset{t\rightarrow}{\boldsymbol{H}}$ in Eq. (33) while the outliers, i.e., Fig.4 (*c*) are modeled by the sparse matrix.

As expected, these convolutive bases have reflected the temporal dynamic characteristics of the analyzed signals. In Fig.5, it is easy to observe that the learned bases match well with the temporal dynamic structure of the sine-like signal. The results of the learned dictionaries with temporal spans 8 and 16 are shown in Fig.5 (*a*) and Fig.5 (*b*), respectively.

In the case of $T = 8$ (as shown in Fig.5 (*a*)), each basis captures the half periodic structure and two out of five bases are kept non-trivial when the algorithm eventually convergences. Besides, since the two patterns have almost the same probability of appearance, their relevance values are thus almost equal.

For comparison, in the case of $T = 16$, one convolutive basis alone is sufficient to represent the periodic structure. Therefore, the remaining four relevance values and their corresponding bases are driven to small positive values close to zeros when the algorithm converges, as seen in the plotted lines of Fig.5 (*b*).

Note that when the basis' temporal span is set to 16, there are plenty of representations that capture the temporal richness. It is even reasonable to expect the same result as in the case of $T = 8$, i.e., the signal is represented by two bases where each basis captures half the periodic structure within 8 time frames, whilst the remaining 8 time slices are padded with zero. However, this case will lead to an expensive objective function due to failed to get a parsimonious representation. Taking this into consideration, the experimental results actually reveal that our algorithm is able to automatic select the most parsimonious representation among all solutions.

This group of experiments reveal that CRHL-NMF is capable of capturing temporal structures, as well as maintaining the merits of RHL-NMF, such as robust to outliers and preferring a parsimonious representations. A theoretical comparison of the two proposed algorithms and the state-of-the-art methods are presented in Table 3. In Table 3, four properties are compared among the algorithms. Parsimonious representation indicates that if the algorithm has the ability to capture the labent structure of the dataset by using the smallest number of bases. Robustness to outliers reveals that if the algorithm is capable to extract the latent component in the presence of arbitrarily large outliers. Modeling temporal continuity shows if the algorithm has the ability of modeling temporal structures in time series data. Hierarchical representation means if the hyper-parameter of the algorithm can be optimized in a hyper-parametric framework.

We have illustrated the efficiency of the proposed algorithms through intuitive experiments on synthetic datasets. Subsequently, we report some real-world applications and evaluate the performance of RHL-NMF and CRHL-NMF.

TABLE 3. Comparison on the properties of various algorithms.

Comparison Algorithms	Parsimonious Represen-tation	Robustness to Outliers	Modeling Temporal Continuity	Hierarchical Represen-tation
NMF	×	×	×	×
ARD-NMF	✓	×	×	✓
CNMF	×	×	✓	×
RNMF	×	✓	×	×
RHL-NMF	✓	✓	×	✓
CRHL-NMF	✓	✓	✓	✓

C. EXPERIMENTS ON REAL WORLD APPLICATIONS

In this subsection, we will present experimental results on real world applications. We first consider a background modeling task which aims to reveal the correlation between surveillance video frames, video background and foreground moving objects.

We apply the algorithm on the video sequence Hall which contains 200 video frames of resolution 144 × 176 from the *i2r* dataset [44]. The parameters are chosen as $a = 10$, $b = 10, \gamma = 1, \kappa = 1$. We run RHL-NMF or CRHL-NMF for 5000 iterations. The estimated background and foreground parts are reconstructed from the columns in the low-rank product and sparse matrices.

The yielded results in Fig.6 demonstrate that the proposed algorithm precisely separates the invariant background and also discovers the optical flow. In Fig.6, the left column, i.e. $\boldsymbol{V}$ in Eq. (10), contains a series of pictures taken in the hall of a business building. The middle column is the low-rank approximation, i.e. $\boldsymbol{WH}$ in Eq. (10). This part is used for background modeling. The third column is the sparse matrix that accommodate for outliers, i.e., $\boldsymbol{S}$ in Eq. (10). This part is used for the optical flow tacking.

Another group of experiments dealing with the separation of speech and background noise is present in Fig.7. In those experiments, our aim is to separate speech from the background noise with repeating patterns in time and frequency. The clean speech file is randomly chosen from the NOIZEUS dataset, [45] while the background noise '*siren*' is one of the 165 natural sounds in [46]. Both files are resampled to 16 kHz and they are mixed at 0 dB SNR. The corresponding magnitude spectrum is shown in the top panel in Fig.7. The parameters are set to $a = 10$, $b = 100$, $\gamma = 1.5$, and $\kappa = 1$. Wiener filtering is adopted for the post processing of the magnitude spectrogram.

As we can see from Fig.7 (*a*) that the background noise '*siren*' demonstrates a strong periodically time-frequency structure and there is a large overlap between speech and noise, which makes the separation task challenging.

Previous study has revealed that sparse and low-rank matrix decomposition can be adopted for unsupervised speech enhancement, [47]. However, in the current experiment, because of the rich time-frequency structure within each frequency bin, the background noise cannot be modeled directly using a simple low-rank model. CRHL-NMF is able to capture the temporal structure, thanks to the temporal duration of its bases. With convolutive bases, the representation



FIGURE 6. Results on background modeling and optical flow tracking of a 50-frame surveillance video sequence from the Hall dataset with resolution 144 × 176. The left column contains pictures taken in the hall of a business building, the middle column is the low-rank approximation (i.e. *WH* in Eq. (10) which is the background modeling) whilst the third column is the sparse matrix that accommodate for outliers (i.e. *S* in Eq. (10) which is the optical flow tacking). Our algorithm has successfully seperated foreground from background.

of the whole spectrum becomes low-rank again, so that the sparse and low-rank matrix decomposition is feasible for speech separation.

To cooperate with the implementation of CRHL-NMF, we first use the technique proposed in [48] to identify the repeating period. Then, we set the range of convolutive bases equal to the length of the repeating period and run CRHL-NMF, which is similar to [37]. The separated spectrum of speech and background noise are presented in Fig.7 (*b*) (i.e., the sparse matrix that accommodates for outliers) and Fig.7 (*c*) (i.e., the reconstructed low-rank component using convolutive basis), respectively. As we can see from the results, CRHL-NMF successfully separate speech from the background '*siren*' noise.

VII. CONCLUSION

In this paper, we have presented a robust hierarchical learning approach for NMF that can perform automatic model order selection in the presence of noise and sparse outliers. We have derived updating algorithms and have shown that the cost function decreases at every updating step. We have also

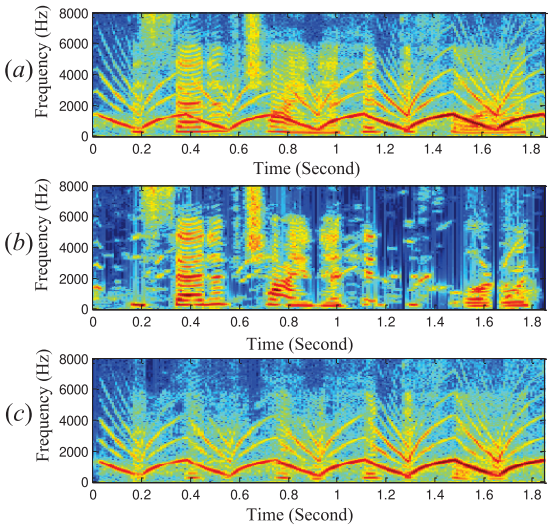


FIGURE 7. The top panel (*a*) shows the spectrogram of the mixture which mixes clean speech with '*siren*' noise at SNR 0 dB. The bottom panel (*c*) demonstrates the reconstructed low-rank component using convolutive basis whilst the middle panel (*b*) is the sparse matrix that accommodates for outliers (i.e., the sparse and varied speech signal). Our algorithm has successfully separated speech from noises.

considered incorporating convolutive bases to improve the representation ability of low-rank modeling. Experimental results in several datasets have shown the efficiency of the proposed algorithms.

REFERENCES

[1] V. Renkens and H. Van Hamme, "Automatic relevance determination for nonnegative dictionary learning in the Gamma-Poisson model," *Signal Process.*, vol. 132, pp. 121–133, Mar. 2017.

[2] C. Févotte, N. Bertin, and J.-L. Durrieu, "Nonnegative matrix factorization with the Itakura–Saito divergence: With application to music analysis," *Neural Comput.*, vol. 21, no. 3, pp. 793–830, 2009.

[3] P. Smaragdis, C. Févotte, G. J. Mysore, N. Mohammadiha, and M. Hoffman, "Static and dynamic source separation using nonnegative factorizations: A unified view," *IEEE Signal Process. Mag.*, vol. 31, no. 3, pp. 66–75, May 2014.

[4] D. D. Lee and H. S. Seung, "Learning the parts of objects by non-negative matrix factorization," *Nature*, vol. 401, no. 6755, p. 788, 1999.

[5] T. Virtanen, J. F. Gemmeke, B. Raj, and P. Smaragdis, "Compositional models for audio processing: Uncovering the structure of sound mixtures," *IEEE Signal Process. Mag.*, vol. 32, no. 2, pp. 125–144, Mar. 2015.

[6] L. Du, X. Li, and Y.-D. Shen, "Robust nonnegative matrix factorization via half-quadratic minimization," in *Proc. IEEE 12th Int. Conf. Data Mining (ICDM)*, Dec. 2012, pp. 201–210.

[7] R. Zhao and V. Y. F. Tan, "Online nonnegative matrix factorization with outliers," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Mar. 2016, pp. 2662–2666.

[8] N. Mohammadiha, P. Smaragdis, and A. Leijon, "Supervised and unsupervised speech enhancement using nonnegative matrix factorization," *IEEE Trans. Audio, Speech, Language Process.*, vol. 21, no. 10, pp. 2140–2151, Oct. 2013.

[9] N. Mohammadiha, P. Smaragdis, G. Panahandeh, and S. Doclo, "A state-space approach to dynamic nonnegative matrix factorization," *IEEE Trans. Signal Process.*, vol. 63, no. 4, pp. 949–959, Feb. 2015.

[10] V. Y. Tan and C. Févotte, "Automatic relevance determination in nonnegative matrix factorization with the β-divergence," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 7, pp. 1592–1605, Jul. 2013.

[11] V. Y. F. Tan and C. Févotte, "Automatic relevance determination in nonnegative matrix factorization," in *Proc. Signal Process. Adapt. Sparse Struct. Represent. (SPARS)*, 2009, pp. 1–5.

[12] A. Lefèvre, F. Bach, and C. Févotte, "Itakura-Saito nonnegative matrix factorization with group sparsity," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, May 2011, pp. 21–24.

[13] A. T. Cemgil, "Bayesian inference for nonnegative matrix factorisation models," *Comput. Intell. Neurosci.*, vol. 2009, pp. 1–17, Feb. 2009, Art. no. 785152.

[14] M. Kim and P. Smaragdis, "Efficient model selection for speech enhancement using a deflation method for nonnegative matrix factorization," in *Proc. IEEE Global Conf. Signal Inf. Process. (GlobalSIP)*, Dec. 2014, pp. 537–541.

[15] D. J. C. MacKay, "Probable networks and plausible predictions—A review of practical Bayesian methods for supervised neural networks," *Netw., Comput. Neural Syst.*, vol. 6, no. 3, pp. 469–505, 1995.

[16] S. D. Babacan, M. Luessi, R. Molina, and A. K. Katsaggelos, "Sparse Bayesian methods for low-rank matrix estimation," *IEEE Trans. Signal Process.*, vol. 60, no. 8, pp. 3964–3977, Aug. 2012.

[17] M. Sun, X. Zhang, and H. Van Hamme, "A stable approach for model order selection in nonnegative matrix factorization," *Pattern Recognit. Lett.*, vol. 54, pp. 97–102, Mar. 2015.

[18] L. Zhang, Z. Chen, M. Zheng, and X. He, "Robust non-negative matrix factorization," *Frontiers Elect. Electron. Eng. China*, vol. 6, no. 2, pp. 192–200, Jun. 2011.

[19] D. Kong, C. Ding, and H. Huang, "Robust nonnegative matrix factorization using $\ell_{2,1}$-norm," in *Proc. 20th ACM Int. Conf. Inf. Knowl. Manage.*, 2011, pp. 673–682.

[20] H. Gao, F. Nie, W. Cai, and H. Huang, "Robust capped norm nonnegative matrix factorization: Capped norm NMF," in *Proc. 24th ACM Int. Conf. Inf. Knowl. Manage.*, 2015, pp. 871–880.

[21] T. Liu and D. Tao, "On the performance of manhattan nonnegative matrix factorization," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 27, no. 9, pp. 1851–1863, Sep. 2016.

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

[23] Y. Li, X. Zhang, M. Sun, G. Min, and J. Yang, "Adaptive extraction of repeating non-negative temporal patterns for single-channel speech enhancement," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Mar. 2016, pp. 494–498.

[24] M. Sun, Y. Li, J. F. Gemmeke, and X. Zhang, "Speech enhancement under low SNR conditions via noise estimation using sparse and low-rank NMF with Kullback–Leibler divergence," *IEEE Trans. Audio, Speech, Language Process.*, vol. 23, no. 7, pp. 1233–1242, Jul. 2015.

[25] P. Sprechmann, A. M. Bronstein, and G. Sapiro, "Learning efficient sparse and low rank models," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 9, pp. 1821–1833, Sep. 2015.

[26] Y. Li, X. Zhang, and M. Sun, "Robust non-negative matrix factorization with β-divergence for speech separation," *ETRI J.*, vol. 39, no. 1, pp. 21–29, 2017.

[27] E. J. Candès, X. Li, Y. Ma, and J. Wright, "Robust principal component analysis?" *J. ACM*, vol. 58, no. 3, p. 11, May 2011.

[28] T. Virtanen, "Monaural sound source separation by nonnegative matrix factorization with temporal continuity and sparseness criteria," *IEEE Trans. Audio, Speech, Language Process.*, vol. 15, no. 3, pp. 1066–1074, Mar. 2007.

[29] G. Mysore and M. Sahani. (Jun. 2012). "Variational inference in non-negative factorial hidden Markov models for efficient audio source separation." [Online]. Available: https://arxiv.org/abs/1206.6468

[30] C. Févotte, "Majorization-minimization algorithm for smooth itakura-saito nonnegative matrix factorization," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, May 2011, pp. 1980–1983.

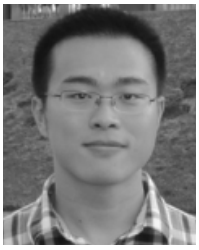
[31] K. W. Wilson, B. Raj, and P. Smaragdis, "Regularized non-negative matrix factorization with temporal dependencies for speech denoising," in *Proc. 9th Annu. Conf. Int. Speech Commun. Assoc.*, 2008, pp. 411–414.

[32] M. Kim and P. Smaragdis, "Single channel source separation using smooth nonnegative matrix factorization with Markov random fields," in *Proc. IEEE Int. Workshop Mach. Learn. Signal Process. (MLSP)*, Sep. 2013, pp. 1–6.

[33] P. Smaragdis, "Convolutive speech bases and their application to supervised speech separation," *IEEE Trans. Audio, Speech, Language Process.*, vol. 15, no. 1, pp. 1–12, Jan. 2007.

[34] D. Wang, R. Vipperla, N. Evans, and T. F. Zheng, "Online non-negative convolutive pattern learning for speech signals," *IEEE Trans. Signal Process.*, vol. 61, no. 1, pp. 44–56, Jan. 2013.

[35] W. Wang, A. Cichocki, and J. A. Chambers, "A multiplicative algorithm for convolutive non-negative matrix factorization based on squared Euclidean distance," *IEEE Trans. Signal Process.*, vol. 57, no. 7, pp. 2858–2864, Jul. 2009.
[36] P. D. O'Ogrady and B. A. Pearlmutter, "Discovering convolutive speech phones using sparseness and non-negativity," in *Proc. Int. Conf. Indepen. Compon. Anal. Signal Separat.* Berlin, Germany: Springer, 2007, pp. 520–527.
[37] Y. Li, X. Zhang, M. Sun, C. Jia, and X. Zou, "Automatic model order selection for convolutive non-negative matrix factorization," *IEICE Trans. Fundam. Electron., Commun. Comput. Sci.*, vol. E99-A, no. 10, pp. 1867–1870, 2016.
[38] U. Şimşekli, T. Virtanen, and A. T. Cemgil, "Non-negative tensor factorization models for Bayesian audio processing," *Digit. Signal Process.*, vol. 47, pp. 178–191, Dec. 2015.
[39] R. Zhao, V. Y. F. Tan, and H. Xu, "Online nonnegative matrix factorization with general divergences," in *Proc. Int. Conf. Artif. Intell. Statist. (AISTATS)*, 2017, pp. 37–45.
[40] F. Sha, Y. Lin, L. K. Saul, and D. D. Lee, "Multiplicative updates for nonnegative quadratic programming," *Neural Comput.*, vol. 19, no. 8, pp. 2004–2031, 2007.
[41] F. Sha, L. K. Saul, and D. D. Lee, "Multiplicative updates for nonnegative quadratic programming in support vector machines," in *Proc. Adv. Neural Inf. Process. Syst.*, 2003, pp. 1065–1072.
[42] D. Donoho and V. Stodden, "When does non-negative matrix factorization give a correct decomposition into parts?" in *Proc. Adv. Neural Inf. Process. Syst.*, 2004, pp. 1141–1148.
[43] V. Renkens, V. Tomar, and H. Van Hamme, "Incrementally learn the relevance of words in a dictionary for spoken language acquisition," in *Proc. SLT*, 2016, pp. 144–150.
[44] L. Li, W. Huang, I. Y.-H. Gu, and Q. Tian, "Statistical modeling of complex backgrounds for foreground object detection," *IEEE Trans. Image Process.*, vol. 13, no. 11, pp. 1459–1472, Nov. 2004.
[45] P. C. Loizou, *Speech Enhancement: Theory and Practice*. Boca Raton, FL, USA: CRC Press, 2007.
[46] S. Norman-Haignere, N. G. Kanwisher, and J. H. McDermott, "Distinct cortical pathways for music and speech revealed by hypothesis-free voxel decomposition," *Neuron*, vol. 88, no. 6, pp. 1281–1296, 2015.
[47] C. Sun, Q. Zhu, and M. Wan, "A novel speech enhancement method based on constrained low-rank and sparse matrix decomposition," *Speech Commun.*, vol. 60, pp. 44–55, May 2014.
[48] Z. Rafii and B. Pardo, "REpeating pattern extraction technique (REPET): A simple method for music/voice separation," *IEEE Trans. Audio, Speech, Language Process.*, vol. 21, no. 1, pp. 73–84, Jan. 2013.



YINAN LI received the Ph.D. degree from Army Engineering University, Nanjing, China. He is currently a Researcher with Space Engineering University, Beijing. His research interests include speech enhancement, pattern recognition, and remote sensing.



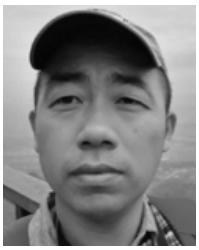
MENG SUN received the Ph.D. degree from the Department of Electrical Engineering (ESAT), KU Leuven. He is currently an Associate Professor with Army Engineering University, China. His research interests include speech processing, unsupervised/semi-supervised machine learning, and sequential pattern recognition.



HUGO VAN HAMME (M'92–SM'11) received the master's degree in engineering (burgerlijk ingenieur) from Vrije Universiteit Brussel (VUB), Brussels, Belgium, in 1987, the M.Sc. degree from Imperial College London, London, U.K, in 1988, and the Ph.D. degree in electrical engineering from VUB, in 1992. From 1993 to 2002, he was with LH Speech Products and ScanSoft, initially as a Senior Researcher and later as a Research Manager. Since 2002, he has been a Professor with the Department of Electrical Engineering, KU Leuven, Leuven, Belgium. His main research interests include automatic speech assessment, assistive speech technology, source separation, and noise robust speech recognition, models of language acquisition, and computer-assisted learning.



XIONGWEI ZHANG received the Ph.D. degree from the PLA University of Science and Technology, China. He is currently a Professor with Army Engineering University. His research interests include speech coding, speech enhancement, and image processing.



JIBIN YANG received the Ph.D. degree from the PLA University of Science and Technology, China. He is currently an Associate Professor with Army Engineering University. His research interests include speech coding, speech recognition, and image processing.