sensors

MDPI

Article

Unsupervised Learning for Monaural Source Separation Using Maximization–Minimization Algorithm with Time–Frequency Deconvolution †

Wai Lok Woo [1,*], Bin Gao [2], Ahmed Bouridane [3], Bingo Wing-Kuen Ling [4] and Cheng Siong Chin [5]

[1] School of Electrical and Electronic Engineering, Newcastle University, Newcastle upon Tyne NE1 7RU, UK
[2] School of Automation Engineering, University of Electronic Science and Technology of China, Chengdu 610054, China; bin_gao@uestc.edu.cn
[3] Department of Computer and Information Sciences, Northumbria University, Newcastle upon Tyne NE1 8ST, UK; ahmed.bouridane@northumbria.ac.uk
[4] Faculty of Information Engineering, Guangdong University of Technology, Guangzhou 510006, China; yongquanling@gdut.edu.cn
[5] Faculty of Science Agriculture and Engineering, Newcastle University, Singapore 599493, Singapore; cheng.chin@ncl.ac.uk
* Correspondence: lok.woo@ncl.ac.uk

† This paper is an extended version of Yu, K.; Woo, W.L.; Dlay, S.S. Variational regularized two-dimensional nonnegative matrix factorization with the flexible β4-divergence for single channel source separation. In Proceedings of the 2nd IET International Conference in Intelligent Signal Processing (ISP), London, UK, 1–2 December 2015.

Received: 9 April 2018; Accepted: 24 April 2018; Published: 27 April 2018

Abstract: This paper presents an unsupervised learning algorithm for sparse nonnegative matrix factor time–frequency deconvolution with optimized fractional β-divergence. The β-divergence is a group of cost functions parametrized by a single parameter β. The Itakura–Saito divergence, Kullback–Leibler divergence and Least Square distance are special cases that correspond to $\beta = 0$, 1, 2, respectively. This paper presents a generalized algorithm that uses a flexible range of β that includes fractional values. It describes a maximization–minimization (MM) algorithm leading to the development of a fast convergence multiplicative update algorithm with guaranteed convergence. The proposed model operates in the time–frequency domain and decomposes an information-bearing matrix into two-dimensional deconvolution of factor matrices that represent the spectral dictionary and temporal codes. The deconvolution process has been optimized to yield sparse temporal codes through maximizing the likelihood of the observations. The paper also presents a method to estimate the fractional β value. The method is demonstrated on separating audio mixtures recorded from a single channel. The paper shows that the extraction of the spectral dictionary and temporal codes is significantly more efficient by using the proposed algorithm and subsequently leads to better source separation performance. Experimental tests and comparisons with other factorization methods have been conducted to verify its efficacy.

Keywords: adaptive signal processing; blind source separation; sensors signal processing; machine learning; maximization–minimization algorithm; β-divergence; matrix deconvolution

1. Introduction

Blind source separation (BSS) [1–8] is an ill-posed problem that cannot be totally solved without some prior information. This entails a certain number of assumptions have to be imposed to render the problem solvable such as channel type (linear [1] versus nonlinear [2]), mutual statistical independence among the sources [3], the number of sources [4], how the sources are mixed (instantaneous [5] versus convolutive [6]), and the location of the sources with respect to the microphones. Several recent solutions have been developed to mitigate some of these constraints. In the work [7], it was previously shown that non-Gaussian stationary process can be approximated as non-stationary Gaussian process which enabled separation involving mixtures of non-Gaussian sources. Of similar concept, a method is proposed for separation by decorrelating multiple non-stationary stochastic sources using a multivariable crosstalk-resistant adaptive noise canceller [8]. In a related method, the problem of speech quality enhancement is tackled using adaptive and non-adaptive filtering algorithms [9]. A two-microphone Gauss–Seidel pseudo affine projection algorithm combined with forward blind source separation is proposed. A higher efficiency in speech enhancement in noisy environment has been attained. The paper [10] proposes rational polynomial functions to replace the original score functions used in standard independent component analysis (ICA). The rational polynomials are derived by the Pade approximant from Taylor series expansion of the original nonlinearities which can be quickly evaluated to enable large-scale multidimensional sets of data characterized by super-Gaussian distribution to be separated within a short period of time. Recently, a bi-variate empirical mode decomposition algorithm combined with complex ICA by entropy bound minimization technique is proposed for convolutive signal separation [11]. In telecommunication problems, neither the direction of arrival (DOA) nor a training sequence is assumed to be available at the receiver. The only assumption is that the transmitted signals satisfy the constant modulus property. In the work [12], a multistage space–time equalizer is proposed to blindly separate signals received by an antenna array from different sources simultaneously. In the algorithm, each stage consists of an adaptive beamformer, a DOA estimator and an equalizer which are jointly optimized using the constant modulus property of the sources. Other than statistical independence and non-Gaussianity, signal separation approach based on second-order statistics of the speech signals using canonical correlation approach [13] has also been proposed. The work [14] considers complex-valued mixing matrix estimation and direction-of-arrival estimation of synchronous orthogonal frequency hopping signals in the underdetermined blind source separation (UBSS). A mixing matrix estimation algorithm is proposed by detecting single source points where only one source contributes its power. While traditional algorithms are usually applied in the ideal sparse environment, the work [15] proposes a solution where multiple input multiple output mixed signals are insufficiently sparse in both time and frequency domains under noisy conditions. The work [16] demonstrates the application of UBSS in addresses the mixing of pipe abrasive debris problem and focuses on the superimposed abrasive debris separation of a radial magnetic field abrasive sensor. Through accurately separating and calculating the morphology and amount of the abrasive debris, the abrasive sensor has provided the system with wear trend and sizes estimation of the wear particles.

In recent years, an alternate class of solutions for BSS based on nonnegative matrix factorization (NMF) [17] has been proposed. Compared to ICA, NMF gives a more part based decomposition and the decomposition is unique under certain conditions, making it unnecessary to impose the constraints in the form of orthogonality and independence [18]. These properties have led to a significant interest in NMF lately for its application in areas of BSS [5,19–24], pattern recognition [25], and dimensionality reduction [26]. Multiplicative update-based families of parameterized cost functions such as the Csiszar's divergences [27,28] were also presented. The NMF is a matrix decomposition technique. Let the data matrix **V** be a nonnegative matrix of dimensions $I \times J$. The aim of NMF is to find two matrices **W** and **H** such that:

$$\mathbf{V} = \mathbf{W}\mathbf{H} \tag{1}$$

or in scalar form,

$$V_{i,j} = \sum_{k} W_{i,k} H_{k,j} \tag{2}$$

where i = 1, 2, ..., $k = 1,\ 2, \ldots,\ K$, and $j = 1,\ 2,\ \ldots,\ J$. When **W** and **H** are nonnegative matrices of dimensions $I \times K$ and $K \times J$, then is usually chosen such that

$$I \times K + K \times J \ll I \times J \tag{3}$$

A sparseness constraint can be added to the cost function [26–31], and this can be achieved by regularization using the L_1-norm leading to Sparse NMF (SNMF). Here, "sparseness" refers to a representational scheme where only a few units (out of a large population) are effectively used to represent typical data vectors. In effect, this implies most units taking values close to zero while only few take significantly non-zero values. Several other types of prior distribution over **W** and **H** can be defined, e.g., it is assumed that the prior of **W** and **H** satisfy the exponential density and the prior for the noise variance is chosen as an inverse gamma density [27]. In the work [28], Gaussian distributions are chosen for both **W** and **H**. The model parameters and hyper parameters are adapted by using the Markov chain Monte Carlo (MCMC) [32]. In all cases, a fully Bayesian treatment is applied to approximate inference for both model parameters and hyper parameters. While these approaches increase the accuracy of matrix factorization, it only works efficiently when a large sample dataset is available. Moreover, it consumes significantly high computational complexity at each iteration to adapt the parameters and its hyper parameters. The NMF with the β-divergence has been previously used in music signal processing [33,34]. In our previous paper [35], we investigated β-divergence for source separation problem. It was shown that improved performance has been attained over integer-based β-divergence. Thus, this motivates research of using β-divergence for music signal processing and source separation. However, all of these works fixed β to some constant values within 0–2, and have not presented any method to determine the desired β value. This significantly constrains the performance of matrix factorization and its ability in separating mixed sources. In addition, these works do not consider the issue of sparsity of the temporal codes which would undermine the quality of matrix factorization when the β value is inappropriately chosen. The selection of the β value should consider the sparseness constraint used in the cost function.

Regardless of the cost function and sparseness constraint being used, the standard NMF or SNMF models are only satisfactory for solving source separation provided that the spectral frequencies of the analyzed audio signal do not change over time. However, this is not the case for many realistic signals such as music and speech. As a result, the spectral dictionary obtained via the NMF or SNMF decomposition is not adequate to capture the temporal dependency of the frequency patterns within the signal. To remedy the situation, a pragmatic approach is to work on a more holistic model based on matrix factor deconvolution [21–24]. In this paper, we work with NMF model extended to two-dimensional time–frequency deconvolution of **W** and **H** where (**W**, **H**) are considered as the matrix factors [22]. Mathematically, this is expressed as

$$\begin{gathered} \mathrm{V}_{i,j} = \sum_{k=1}^{K} \sum_{\tau=0}^{\tau_{max}} \sum_{\phi=0}^{\phi_{max}} W^{\tau}_{i-\phi,k} H^{\phi}_{k,j-\tau} \\ \mathbf{V} = \sum_{\tau=0}^{\tau_{max}} \sum_{\phi=0}^{\phi_{max}} \overset{\downarrow\phi}{\mathbf{W}^{\tau}} \overset{\rightarrow\tau}{\mathbf{H}^{\phi}} \end{gathered} \tag{4}$$

where i and i represent the frequency and time index, respectively, k indicates the factor number, τ represents the temporal shift and ϕ is the frequency shift. The terms τ_{max} and ϕ_{max} are the maximum temporal and frequency shift, respectively. With this definition, both $W^{\tau}_{i,k}$ and $H^{\phi}_{k,j}$ have tensorial structures with dimension $I \times K \times \tau_{max}$ and $K \times J \times \phi_{max}$, respectively. Thus, $W^{\tau}_{i,k}$ represents the τth-slice of the kth-spectral basis while $H^{\phi}_{k,j}$ represents the associated ϕth-slice of the kth-temporal code. The downward and rightward arrow signs denote the corresponding shifting direction of each column in $\mathbf{W}^{\tau}$ and each row in $\mathbf{H}^{\phi}$ by the amount indicated by τ and ϕ, respectively.

Model (4) represents both temporal structure and the pitch change which occur when an instrument plays different notes. In the log-frequency spectrogram, the pitch change corresponds to a displacement on the frequency axis. Where previous NMF methods needed one component to model each note for each instrument, Model (4) represents each instrument compactly by a single time–frequency profile convolved in both time and frequency by a time–pitch weight matrix. This model dramatically decreases the number of components needed to model various instruments and effectively solves the blind single channel source separation problem for certain classes of musical signals. When polyphonic music is modeled by factorizing the magnitude spectrogram with NMF, each instrument is modeled by an instantaneous frequency signature which can vary over time. However, the NMF requires multiple basis functions to represent tones with different pitch values. The two-dimensional time–frequency deconvolution model implicitly solves the problem of grouping notes. Thus, all notes for an instrument is an identical pitch shifted time–frequency signature, Model (4) will give better estimates of these signatures, because more examples of different notes are used to compute each time–frequency signature. In the event when this assumption does not hold, it might still hold in a region of notes for an instrument. Furthermore, the two-dimensional time–frequency deconvolution model can explain the spectral differences between two notes of different pitch by the two-dimensional deconvolution of the time–frequency signature.

The novelty of this paper can be summarized as follows: Firstly, a new algorithm is developed for sparse nonnegative matrix factor time–frequency deconvolution optimized with fractional β-divergence. Secondly, the maximization–minimization algorithm is developed to derive the auxiliary cost function which caters for any β value. The paper shows that the optimal β that leads to the desired performance is not necessarily limited to the special cases of integer β but extends to fractional values. Thirdly, it is analytically shown that the convergence of the proposed algorithm is guaranteed under the auxiliary function. Fourthly, a method is proposed to estimate the fractional β within the context of monoaural source separation. Finally, the paper proposes an adaptive method to estimate the sparsity parameter for each of the individual temporal code.

The remainder of the paper is organized as follows: In Section 2, the new algorithm for matrix factor time–frequency deconvolution model with β-divergence based on the maximization–minimization algorithmic framework is derived. Real application of blind source separation using the proposed method and comparisons with other matrix factorization methods are presented in Section 3. Finally, Section 4 concludes the paper.

2. Background

2.1. *β-Divergence Cost Function*

The NMF problem can be written as the minimization of an objective function:

$$\mathrm{D}(\mathbf{V}|\mathbf{WH}) = \sum_{i,j} d_\beta\left(V_{i,j}|\Lambda_{i,j}\right) \tag{5}$$

The general β-divergence [24,31] is defined as:

$$d_\beta(y|x) = \begin{cases} \frac{y^\beta}{\beta(\beta-1)} + \frac{x^\beta}{\beta} - \frac{yx^{\beta-1}}{\beta-1}, & \beta \to \mathbb{R}/\{0,1\} \\ y(\log y - \log x) + (x-y), & \beta = 1 \\ \frac{y}{x} - \log\frac{y}{x} - 1 & \beta = 0 \end{cases} \tag{6}$$

when $\beta = 2$, this matches with the first β-divergence and the update algorithm is referred to as the "Least Square" [17]. When we use the second β-divergence with $\beta = 1$, the update algorithm is referred to as the "Kullback–Leibler" [17]. When the third β-divergence with $\beta = 0$ is used, the update algorithm is referred to as the "Itakura–Saito" [33]. These algorithms have their own advantages and disadvantages. If the sources have large dynamic difference in the power, the Itakura–Saito divergence would have better performance than other NMF algorithms. The Least Square and Kullback–Leibler NMFs are more suited when the power of sources are close to other. However, it is difficult to define the difference of power between the sources, and therefore it is difficult to choose the algorithms. In this paper, we present the results to show that the best results are not necessarily limited to the above integer β special cases. The use the fractional β-divergence is expected to yield more realistic and optimized results than the previous NMF algorithms. For completeness of presentation, the following section briefly reviews the update function based on the Least Square and Kullback–Leibler criterion.

2.1.1. Least Square Distance

The Least Square NMF algorithm introduced by Lee and Seung [17] defines the β-divergence as Least Squares divergence when $\beta = 2$. First, we consider the least square cost function:

$$\begin{aligned} C_{LS} = \tfrac{1}{2}||\mathbf{V} - \mathbf{\Lambda}||_F^2 &= \sum_{i,j} d_2(V_{i,j}|\Lambda_{i,j}) \\ &= \frac{1}{2}\sum_{i,j}\left(V_{i,j} - \Lambda_{i,j}\right)^2 \end{aligned} \tag{7}$$

Differentiating C_{LS} with respect to $W^{\tau}_{i,k}$ and $H^{\phi}_{k,j}$, and plugging the multiplicative update algorithm for $\theta = \left\{ (W^{\tau}_{i,k})_{I\times K\times \tau_{max}}, (H^{\phi}_{k,j})_{K\times J\times \phi_{max}} \right\}$:

$$\theta \leftarrow \theta \cdot \left(\frac{\left[\nabla d_\beta(y|x)\right]_-}{\left[\nabla d_\beta(y|x)\right]_+} \right)$$

where $\partial d_\beta(y|x)/\partial\theta = \left[\nabla d_\beta(y|x)\right]_+ - \left[\nabla d_\beta(y|x)\right]_-$, which leads to the following $\mathbf{W}^{\tau}$ and $\mathbf{H}^{\phi}$ updates:

$$\mathbf{W}^{\tau} = \mathbf{W}^{\tau} \cdot \frac{\sum_{\phi} \overset{\uparrow\phi\leftarrow\tau}{\mathbf{V}} \cdot \overset{\rightarrow\tau}{\mathbf{H}^{\phi}}^{T}}{\sum_{\phi} \overset{\uparrow\phi\leftarrow\tau}{\mathbf{\Lambda}} \cdot \overset{\rightarrow\tau}{\mathbf{H}^{\phi}}^{T}} \tag{8}$$

$$\mathbf{H}^{\phi} = \mathbf{H}^{\phi} \cdot \frac{\sum_{\tau} \overset{\downarrow\phi}{\mathbf{W}^{\tau}}^{T} \cdot \overset{\uparrow\phi\leftarrow\tau}{\mathbf{V}}}{\sum_{\tau} \overset{\downarrow\phi}{\mathbf{W}^{\tau}}^{T} \cdot \overset{\uparrow\phi\leftarrow\tau}{\mathbf{\Lambda}}} \tag{9}$$

where "$A \cdot B$" represents element-wise multiplication.

2.1.2. Kullback–Liebler Divergence

When $\beta = 1$, the β-divergence is identical to the Kullback–Leibler divergence. The Kullback–Leibler divergence is expressed as:

$$C_{KL} = \sum_{i,j} d_1\left(V_{i,j}|\Lambda_{i,j}\right) = \sum_{i,j} V_{i,j} \log \frac{V_{i,j}}{\Lambda_{i,j}} - V_{i,j} + \Lambda_{i,j} \tag{10}$$

By following similar steps as the Least Square, we can derive the update function as follow:

$$\mathbf{W}^{\tau} = \mathbf{W}^{\tau} \cdot \frac{\sum_{\phi} \left(\frac{\overset{\uparrow\phi\leftarrow\tau}{\mathbf{V}}}{\mathbf{\Lambda}} \right) \cdot \overset{\rightarrow\tau}{\mathbf{H}^{\phi}}^{T}}{\sum_{\phi} 1 \cdot \phi^{T}} \tag{11}$$

$$\mathbf{H}^{\phi} = \mathbf{H}^{\phi} \cdot \frac{\sum_{\tau} \overset{\downarrow\phi}{\mathbf{W}^{\tau}}^{T} \cdot \left(\frac{\overset{\uparrow\phi\leftarrow\tau}{\mathbf{V}}}{\mathbf{\Lambda}} \right)}{\sum_{\tau} \overset{\downarrow\phi}{\mathbf{W}^{\tau}}^{T} \cdot 1} \tag{12}$$

where "$\frac{A}{B}$ " represents element-wise division and "**1**" is a column vector of unit elements.

2.2. Auxiliary Cost Function of Fractional β-Divergence for Matrix Factors Time–Frequency Deconvolution

In this subsection, we introduce the cost function for the fractional β-divergence matrix factors time–frequency deconvolution model. The algorithm allows the user to choose a fractional β value instead of using the previous NMF algorithms which constrain β to special cases of integer value. After the derivation, this paper shows the steps on how the update function of the fractional β-divergence is obtained for the parameters. Firstly, the first derivative of $d_\beta(y|x)$ are given by

$$d_\beta^{\prime(y|x)} = y^{\beta-2}(y-x) \tag{13}$$

This shows that y is continuous in β and thus the second derivative of $d_\beta(y|x)$ is given by

$$d_\beta^{\prime\prime(y|x)} = y^{\beta-3}[(\beta-1)y-(\beta-2)x] \tag{14}$$

The second derivative shows that the β-divergence is convex for y in $\beta \in [1,2]$. Outside of this range, $d_\beta(y|x)$ can be expressed as:

$$d_\beta(y|x) = \check{d}(y|x) + \hat{d}(y|x) + \bar{d}(x) \tag{15}$$

where $\check{d}(y|x)$ is a convex function of y, $\hat{d}(y|x)$ is a concave function of y, and $\bar{d}(x)$ is a constant of y. Table 1 shows the various functions for $\check{d}(y|x)$, $\hat{d}(y|x)$ and $\bar{d}(x)$. The problem we want to tackle is to minimize the following function with respect to $\theta = \left\{ \left(W_{i,k}^{\tau}\right)_{I\times K\times\tau_{max}}, \left(H_{k,j}^{\phi}\right)_{K\times J\times\phi_{max}} \right\}$ where β can assume fractional number:

$$\begin{aligned} G(\theta) &= \sum_{i,j} d_\beta\left(V_{i,j} \middle| \textstyle\sum_{k=1}^{K} \sum_{\tau=0}^{\tau_{max}} \sum_{\phi=0}^{\phi_{max}} W_{i-\phi,k}^{\tau} H_{k,j-\tau}^{\phi}\right) \\ &= \frac{1}{\beta(\beta-1)} \sum_{i,j} V_{i,j}^{\beta} + \sum_{i,j} \underbrace{\frac{1}{\beta}\left(\sum_{k,\tau\phi} W_{i-\phi,k}^{\tau} H_{k,j-\tau}^{\phi}\right)^{\beta}}_{G_1(\beta)} + \sum_{i,j} V_{i,j} \underbrace{\left(-\frac{1}{\beta-1}\sum_{k,\tau\phi} W_{i-\phi,k}^{\tau} H_{k,j-\tau}^{\phi}\right)^{\beta-1}}_{G_2(\beta)} \end{aligned} \tag{16}$$

Table 1. Differentiable Convex-Concave-Constant Decomposition of β-Divergence.

Range	$\check{d}(y\|x)$	$\hat{d}(y\|x)$	$\bar{d}(x)$
$\beta < 1$ and $\beta \neq 0$	$-\frac{1}{\beta-1}xy^{\beta-1}$	$\frac{1}{\beta}y^{\beta}$	$y^{\beta-1}$
$\beta = 0$	$xy^{\beta-1}$	$\log y$	y^{-1}
$1 \leq \beta \leq 2$	$d_\beta(x\|y)$	0	0
$\beta > 2$	$\frac{1}{\beta}y^{\beta}$	$-\frac{1}{\beta-1}xy^{\beta-1}$	$-xy^{\beta-2}$

In Equation (16), $G_1(\beta)$ is convex for $\beta \geq 1$ and concave for $\beta < 1$, and $G_2(\beta)$ is convex for $\beta \leq 2$ and concave for $\beta > 2$. Thus, there is a need to alleviate this problem by decomposing the above function into several terms to be either convex or concave depending on the value of β and use the appropriate inequalities to build an auxiliary function.

Lemma 1. *For the case of $\beta \geq 1$, we have*

$$\frac{1}{\beta}\left(\sum_{k,\tau\phi} W_{i-\phi,k}^{\tau} H_{k,j-\tau}^{\phi}\right)^{\beta} \leq \frac{1}{\beta}\sum_{k,\tau\phi} \omega_{i,j,k,\tau,\phi}\left(\frac{W_{i-\phi,k}^{\tau} H_{k,j-\tau}^{\phi}}{\omega_{i,j,k,\tau,\phi}}\right)^{\beta} = P_{i,j}^{(\beta)} \tag{17}$$

where $\omega_{i,j,k,\tau,\phi} \geq 0$ *for all* k, τ, ϕ *and* $\sum\limits_{k,\tau\phi} \omega_{i,j,k,\tau,\phi} = 1$. *The equality holds when*

$$\omega_{i,j,k,\tau,\phi} = \frac{W^{\tau}_{i-\phi,k} H^{\phi}_{k,j-\tau}}{\sum_{k',\tau',\phi'} W^{\tau'}_{i-\phi',k'} H^{\phi'}_{k',j-\tau'}} \tag{18}$$

Proof. Let $f : \mathbb{R} \to \mathbb{R}$ be a convex function. If $\alpha_k (k = 1, 2, \ldots, K)$ satisfies $\forall k, \alpha_k > 0$ and $\sum\limits_k \alpha_k = 1$, then for any $x_k (k = 1, 2, \ldots, K) \in \mathbb{R}$,

$$f\left(\sum_k x_k\right) \leq \sum_k \alpha_k f\left(\frac{x_k}{\alpha_k}\right)$$

and with equality holds if and only if $\alpha_k = x_k / \sum\limits_k x_k$. Substituting $f(\cdot) = \frac{1}{\beta}(\cdot)^\beta$ with $\beta \geq 1$, $x_k = W^{\tau}_{i-\phi,k} H^{\phi}_{k,j-\tau}$ and $\alpha_k = \omega_{i,j,k,\tau,\phi}$ yields Equation (16). □

Lemma 2. *For the case of* $\beta < 1$, *we have*

$$\frac{1}{\beta}\left(\sum_{k,\tau\phi} W^{\tau}_{i-\phi,k} H^{\phi}_{k,j-\tau}\right)^{\beta} \leq \Lambda^{\beta-1}_{i,j}\left(\sum_{k,\tau\phi} W^{\tau}_{i-\phi,k} H^{\phi}_{k,j-\tau} - \Lambda_{i,j}\right) + \frac{\Lambda^{\beta}_{i,j}}{\beta} = Q^{(\beta)}_{i,j} \tag{19}$$

The equality holds when

$$\Lambda_{i,j} = \sum_{k,\tau\phi} W^{\tau}_{i-\phi,k} H^{\phi}_{k,j-\tau} \tag{20}$$

Proof. Let $f : \mathbb{R} \to \mathbb{R}$ be a continuously differentiable and concave function. Then, for any point z,

$$f(x) \leq f'(x)(x - z) + f(z)$$

and with equality holds if and only if $x = z$. Substituting $f(\cdot) = \frac{1}{\beta}(\cdot)^\beta$ with $\beta < 1$, $x = \sum\limits_{k,\tau\phi} W^{\tau}_{i-\phi,k} H^{\phi}_{k,j-\tau}$ and $z = \Lambda_{i,j}$ yields Equation (17). □

Using Lemmas 1 and 2, we may proceed with the following analysis. When $\beta < 1$, we use $Q^{(\beta)}_{i,j}$ instead of $G_1(\beta)$ and $V_{i,j} P^{(\beta-1)}_{i,j}$ instead of $G_2(\beta)$, then the cost function becomes a convex function. Let us denote $G^{+}(\theta|\hat{\theta})$ as an auxiliary function for and $\hat{\theta} = \left\{ \left(\omega_{i,j,k,\tau,\phi}\right)_{I \times J \times K \times \tau_{max} \times \phi_{max}}, \left(\Lambda_{i,j}\right)_{I \times J} \right\}$ as the auxiliary variables. For $G^{+}(\theta|\hat{\theta})$ to qualify as auxiliary function, it must satisfy $G(\theta) = \min\limits_{\hat{\theta}} G^{+}(\theta|\hat{\theta})$. Thus, the cost function can be shown to be bounded by the auxiliary function $G^{+}(\theta|\hat{\theta})$:

$$G(\theta) \leq G^{+}(\theta|\hat{\theta}) = \sum_{i,j} \frac{V^{\beta}_{i,j}}{\beta(\beta-1)} + Q^{(\beta)}_{i,j} - V_{i,j} P^{(\beta-1)}_{i,j} \tag{21}$$

when $1 \leq \beta \leq 2$, we use $P^{(\beta)}_{i,j}$ instead of $G_1(\beta)$ and $V_{i,j} P^{(\beta-1)}_{i,j}$ instead of $G_2(\beta)$, then the cost function becomes a convex function and is bounded by the auxiliary function $G^{+}(\theta|\hat{\theta})$:

$$G(\theta) \leq G^{+}(\theta|\hat{\theta}) = \sum_{i,j} \frac{V^{\beta}_{i,j}}{\beta(\beta-1)} + P^{(\beta)}_{i,j} - V_{i,j} P^{(\beta-1)}_{i,j} \tag{22}$$

Finally, when $\beta > 2$, we use $P_{i,j}^{(\beta)}$ instead of $G_1(\beta)$ and $V_{i,j}Q_{i,j}^{(\beta-1)}$ instead of $G_2(\beta)$, then the cost function is bounded by

$$G(\theta) \leq G^{+}(\theta|\hat{\theta}) = \sum_{i,j} \frac{V_{i,j}^{\beta}}{\beta(\beta-1)} + P_{i,j}^{(\beta)} - V_{i,j}Q_{i,j}^{(\beta-1)} \tag{23}$$

From above, we can conclude that

$$G(\theta) \leq G^{+}(\theta|\hat{\theta}) = \sum_{i,j} \frac{V_{i,j}^{\beta}}{\beta(\beta-1)} + \{ \begin{array}{ll} Q_{i,j}^{(\beta)} - V_{i,j}P_{i,j}^{(\beta-1)}, & (\beta < 1) \\ P_{i,j}^{(\beta)} - V_{i,j}P_{i,j}^{(\beta-1)}, & (1 \leq \beta \leq 2) \\ P_{i,j}^{(\beta)} - V_{i,j}Q_{i,j}^{(\beta-1)}, & (\beta > 2) \end{array} \tag{24}$$

The equality holds when $\hat{\theta}$ satisfies Equations (18) and (20). The above function yields three different sub-functions which depend on the β value. In different β range, we use different cost function in the algorithm. This allows the user to choose the optimal β value to separate the mixture and caters for more flexibility than the previous algorithms.

2.3. Auxiliary Update Function of "Fractional" β-Divergence

To minimize $G^{+}(\theta|\hat{\theta})$, we formulate the derivative of $G^{+}(\theta|\hat{\theta})$ with respect to θ. First, we consider the derivative for $W_{i,k}^{\tau}$:

$$\frac{\partial G^{+}(\theta|\hat{\theta})}{\partial W_{i,k}^{\tau}} = \mathcal{V}_{\mathrm{w}} - \mathcal{W}_{\mathrm{w}} \tag{25}$$

where

$$\mathcal{V}_{\mathrm{w}} = \begin{cases} \sum\limits_{j,\phi} \Lambda_{i,j}^{\beta-1} H_{k,j-\tau}^{\phi}, & (\beta < 1) \\ \left(W_{i,k}^{\tau}\right)^{\beta-1} \sum\limits_{j,\phi} \omega_{i+\phi,j,k,\tau,\phi}^{1-\beta} \left(H_{k,j-\tau}^{\phi}\right)^{\beta}, & (\beta \geq 1) \end{cases} \tag{26}$$

$$\mathcal{W}_{\mathrm{w}} = \begin{cases} (W_{i,k}^{\tau})^{\beta-2} \sum\limits_{j,\phi} V_{i+\phi,j}\omega_{i+\phi,j,k,\tau,\phi}^{2-\beta} \left(H_{k,j-\tau}^{\phi}\right)^{\beta-1} & (\beta \leq 2) \\ \sum\limits_{j,\phi} V_{i+\phi,j}\Lambda_{i+\phi,j}^{\beta-2} H_{k,j-\tau}^{\phi}, & (\beta > 2) \end{cases} \tag{27}$$

The second derivative of $G^{+}(\theta|\hat{\theta})$ with respect to $W_{i,k}^{\tau}$ in then expressed as:

$$\frac{\partial^2 G^{+}(\theta|\hat{\theta})}{\partial W_{i,k}^{\tau} \partial W_{i',k'}^{\tau'}} = (\mathcal{V}_{w}{}' - \mathcal{W}_{w}{}')\delta_{i,i'}\delta_{k,k'}\delta_{\tau,\tau'} \tag{28}$$

where $\delta_{i,j} = 1$ if $i = j$ and $\delta_{i,j} = 0$ if $i \neq j$, and

$$\mathcal{V}_{\mathrm{w}}' = \begin{cases} 0, & (\beta < 1) \\ (\beta-1)\left(W_{i-\phi,k}^{\tau}\right)^{\beta-2} \sum\limits_{j,\phi} \omega_{i+\phi,j,k,\tau,\phi}^{1-\beta} (H_{k,j-\tau}^{\phi})^{\beta}, \ (\beta \geq 1) & (\beta \geq 1) \end{cases} \tag{29}$$

$$\mathcal{W}_{\mathrm{w}}' = \begin{cases} (\beta-2)(W_{i,k}^{\tau})^{\beta-3} \sum\limits_{j,\phi} V_{i+\phi,j}\omega_{i+\phi,j,k,\tau,\phi}^{2-\beta} (H_{k,j-\tau}^{\phi})^{\beta-1}, & (\beta \leq 2) \\ 0, & (\beta > 2) \end{cases} \tag{30}$$

We can see $G(\theta|\hat{\theta})$ is a convex function in $W^{\tau}_{i,k}$, so by setting $\frac{\partial G(\theta|\hat{\theta})}{\partial W_{i,k}}$ to 0, we can then express the update function for $W^{\tau}_{i,k}$ as:

$$W^{\tau}_{i,k} = \begin{cases} \left(\dfrac{\sum_{j,\phi} V_{i+\phi,j}\ \omega^{2-\beta}_{i+\phi,j,k,\tau,\phi} \left(H^{\phi}_{k,j-\tau} \right)^{\beta-1}}{\sum_{j,\phi} \Lambda^{\beta-1}_{i,j} H^{\phi}_{k,j-\tau}} \right)^{\frac{1}{2-\beta}}, & (\beta < 1) \\ \dfrac{\sum_{j,\phi} V_{i+\phi,j}\ \omega^{2-\beta}_{i+\phi,j,k,\tau,\phi} \left(H^{\phi}_{k,j-\tau} \right)^{\beta-1}}{\sum_{j,\phi} \omega^{1-\beta}_{i+\phi,j,k,\tau,\phi} \left(H^{\phi}_{k,j-\tau} \right)^{\beta}}, & (1 \le \beta \le 2) \\ \left(\dfrac{\sum_{j,\phi} V_{i+\phi,j}\ \Lambda^{\beta-2}_{i+\phi,j}\ H^{\phi}_{k,j-\tau}}{\sum_{j,\phi} \omega^{1-\beta}_{i+\phi,j,k,\tau,\phi} \left(H^{\phi}_{k,j-\tau} \right)^{\beta}} \right)^{\frac{1}{\beta-1}}, & (\beta > 2) \end{cases} \tag{31}$$

We next consider the auxiliary variables $\hat{\theta}$. Since both Equations (17) and (18) minimize $G^{+}(\theta|\hat{\theta})$ with respect to $\hat{\theta}$, substituting these into Equation (30) gives the following update rule:

$$W^{\tau}_{i,k} = W^{\tau}_{i,k} \left(\frac{\sum_{j,\phi} V_{i+\phi,j}\ \Lambda^{\beta-2}_{i+\phi,j}\ H^{\phi}_{k,j-\tau}}{\sum_{j,\phi} \Lambda^{\beta-1}_{i+\phi,j}\ H^{\phi}_{k,j-\tau}} \right)^{\delta(\beta)} \tag{32}$$

where

$$\delta(\beta) = \begin{cases} \frac{1}{2-\beta}, & (\beta < 1) \\ 1, & (1 \le \beta \le 2) \\ \frac{1}{\beta-1}, & (\beta > 2) \end{cases} \tag{33}$$

The above can be written in the matrix form:

$$\mathbf{W}^{\tau} = \mathbf{W}^{\tau} \cdot \left[\frac{\sum_{\phi} \left(\overset{\uparrow\phi}{\mathbf{V}} \cdot \overset{\uparrow\phi}{\mathbf{\Lambda}}^{(\beta-2)} \right) \overset{\rightarrow\tau}{\mathbf{H}^{\phi}}^{T}}{\sum_{\phi} \overset{\uparrow\phi}{\mathbf{\Lambda}}^{(\beta-1)} \overset{\rightarrow\tau}{\mathbf{H}^{\phi}}^{T}} \right]^{\delta(\beta)} \tag{34}$$

Similarly, for $H^{\phi}_{k,j}$ update function, first we have

$$\frac{\partial G^{+}(\theta|\hat{\theta})}{\partial H^{\phi}_{k,j}} = \mathcal{V}_{\mathrm{H}} - \mathcal{W}_{\mathrm{H}} \tag{35}$$

where

$$\mathcal{V}_{H} = \begin{cases} \sum\limits_{i,\tau} \Lambda^{\beta-1}_{i,j+\tau}\ W^{\tau}_{i-\phi,k}\ , & (\beta < 1) \\ (H^{\phi}_{k,j})^{\beta-1} \sum\limits_{i,\tau} \omega^{1-\beta}_{i,j+\tau,k,\tau,\phi} (W^{\tau}_{i-\phi,k})^{\beta}\ , & (\beta \ge 1) \end{cases} \tag{36}$$

$$\mathcal{W}_{H} = \begin{cases} \left(H^{\phi}_{k,j} \right)^{\beta-2} \sum\limits_{i,\tau} V_{i,j+\tau}\ \omega^{2-\beta}_{i,j+\tau,k,\tau,\phi} (W^{\tau}_{i-\phi,k})^{\beta-1}\ , & (\beta \le 2) \\ \sum\limits_{i,\tau} V_{i,j+\tau} \Lambda^{\beta-2}_{i,j+\tau} W^{\tau}_{i-\phi,k}\ , & (\beta > 2) \end{cases} \tag{37}$$

From Equations (17)–(31), we can minimize the cost-function by setting $\frac{\partial G_s\left(\theta|\hat{\theta}\right)}{\partial H_{k,j}^{\varphi}} = 0$ and obtain $H_{k,j}^{\phi}$ as:

$$H_{k,j}^{\phi} = \begin{cases} \left(\frac{\sum_{i,\tau} V_{i,j+\tau}\ \omega_{i,j+\tau,k,\tau,\phi}^{2-\beta} \left(W_{i-\phi,k}^{\tau}\right)^{\beta-1}}{\sum_{i,\tau} \Lambda_{i,j+\tau}^{\beta-1}\ W_{i-\phi,k}^{\tau}} \right)^{\frac{1}{2-\beta}} & (\beta < 1) \\ \frac{\sum_{i,\tau} V_{i,j+\tau}\ \omega_{i,j+\tau,k,\tau,\phi}^{2-\beta} \left(W_{i-\phi,k}^{\tau}\right)^{\beta-1}}{\sum_{i,\tau} \omega_{i,j+\tau,k,\tau,\phi}^{1-\beta} \left(W_{i-\phi,k}^{\tau}\right)^{\beta}} & (1 \leq \beta \leq 2) \\ \left(\frac{\sum_{i,\tau} V_{i,j+\tau} \Lambda_{i,j+\tau}^{\beta-2} W_{i-\phi,k}^{\tau}}{\sum_{i,\tau} \omega_{i,j+\tau,k,\tau,\phi}^{1-\beta} \left(W_{i-\phi,k}^{\tau}\right)^{\beta}} \right)^{\frac{1}{\beta-1}}, & (\beta > 2) \end{cases} \tag{38}$$

Again, since both Equations (19) and (20) minimizes $G^{+}(\theta|\hat{\theta})$ with respect to $\hat{\theta}$, substituting these into Equation (38) gives the following update rule for $H_{k,j}^{\phi}$:

$$H_{k,j}^{\phi} = H_{k,j}^{\phi} \left(\frac{\sum_{i,\tau} V_{i,j+\tau}\ \Lambda_{i,j+\tau}^{\beta-2}\ W_{i-\phi,k}^{\tau}}{\sum_{i,\tau} \Lambda_{i,j+\tau}^{\beta-1}\ W_{i-\phi,k}^{\tau}} \right)^{\delta(\beta)} \tag{39}$$

In matrix form, the above can be written as

$$\mathbf{H}^{\phi} = \mathbf{H}^{\phi} \cdot \left[\frac{\sum_{\tau} \overset{\downarrow\phi}{\mathbf{W}^{\tau}}^{T} \left(\overset{\leftarrow\tau}{\mathbf{V}} \cdot \overset{\leftarrow\tau}{\mathbf{\Lambda}}^{(\beta-2)} \right)}{\sum_{\tau} \overset{\downarrow\phi}{\mathbf{W}^{\tau}}^{T}\ \overset{\leftarrow\tau}{\mathbf{\Lambda}}^{(\beta-1)}} \right]^{\delta(\beta)} \tag{40}$$

2.4. Sparsity-Aware Optimization

The cost-function in Equation (21) can be augmented with a regularization term to render sparsity to the solution. We can define a prior on $\mathbf{H}$ as an exponential distribution with independent decay parameters, namely,

$$p(\mathbf{H}|\lambda) = \prod_{\phi} p\left(\mathbf{H}^{\phi}|\boldsymbol{\lambda}^{\phi}\right) = \prod_{\phi}\prod_{k}\prod_{j} p\left(H_{k,j}^{\phi}|\lambda_{k,j}^{\phi}\right) \tag{41}$$

where $p\left(H_{k,j}^{\phi}|\lambda_{k,j}^{\phi}\right) = \prod_{\phi}\prod_{k}\prod_{j} \lambda_{k,j}^{\phi} \exp\left(-\lambda_{k,j}^{\phi} H_{k,j}^{\phi}\right)$. The negative log prior on $\mathbf{H}$ is defined as $-\log p(\mathbf{H}|\boldsymbol{\lambda}) = f(\mathbf{H}) = \sum_{\phi,k,j} \left(\lambda_{k,j}^{\phi} H_{k,j}^{\phi} - \log \lambda_{k,j}^{\phi}\right)$. It is worth pointing out that *each individual element* in $\mathbf{H}$ is constrained to an exponential distribution with independent decay parameter $\lambda_{k,j}^{\phi}$ so that each element in $\mathbf{H}$ can be driven to be optimally sparse in the L_1-norm. Other forms of sparseness exist[19] but the proposed L_1-norm is computationally favourable. First, we define $G_s(\theta)$ and $G_s(\theta|\hat{\theta})$ as follow:

$$\begin{aligned} G_s(\theta) &\triangleq G(\theta) + \alpha f(\mathbf{H}) \\ &\leq G^{+}(\theta|\hat{\theta}) + \alpha f(\mathbf{H}) \\ &= G_s(\theta|\hat{\theta}) \end{aligned} \tag{42}$$

where α is the regularization constant. To avoid the scaling misbehavior when incorporating the sparseness for $\mathbf{H}$, we reformulate the cost function to work with normalized matrix for $\mathbf{W}^{\tau}$ i.e.,

$$\overline{W}_{i,k}^{\tau} = \frac{W_{i,k}^{\tau}}{\sqrt{\sum_{\tau,i}\left(W_{i,k}^{\tau}\right)^{2}}} = \frac{W_{i,k}^{\tau}}{||\mathbf{W}_{k}||_{2}} \tag{43}$$

and

$$\overline{\Lambda}_{i,j} = \sum_{k,\tau,\phi} \overline{W}^{\tau}_{i-\phi,k}\, H^{\phi}_{k,j-\tau} \tag{44}$$

Thus, the cost function takes the following form:

$$\begin{aligned} G_s(\theta) &\leq G_s(\theta|\hat{\theta}) \\ &= \sum_{i,j} \frac{V^{\beta}_{i,j}}{\beta(\beta-1)} + \alpha \sum_{\phi,k,j} \left(\lambda^{\phi}_{k,j} H^{\phi}_{k,j} - \log \lambda^{\phi}_{k,j}\right) + \begin{cases} \overline{Q}^{(\beta)}_{i,j} - V_{i,j}\overline{P}^{(\beta-1)}_{i,j}, & (\beta < 1) \\ \overline{P}^{(\beta)}_{i,j} - V_{i,j}\overline{P}^{(\beta-1)}_{i,j}, & (1 \leq \beta \leq 2) \\ \overline{P}^{(\beta)}_{i,j} - V_{i,j}\overline{Q}^{(\beta-1)}_{i,j}, & (\beta > 2) \end{cases} \end{aligned} \tag{45}$$

where

$$\overline{Q}^{(\beta)}_{i,j} = \overline{\Lambda}^{\beta-1}_{i,j} \left(\sum_{k,\tau\phi} \overline{W}^{\tau}_{i-\phi,k} H^{\phi}_{k,j-\tau} - \overline{\Lambda}_{i,j} \right) + \frac{\overline{\Lambda}^{\beta-1}_{i,j}}{\beta}$$

and

$$\overline{P}^{(\beta)}_{i,j} = \frac{1}{\beta} \sum_{k,\tau,\phi} \omega_{i,j,k,\tau,\phi} \left(\frac{\overline{W}^{\tau}_{i-\phi,k} H^{\phi}_{k,j-\tau}}{\omega_{i,j,k,\tau,\phi}} \right)^{\beta}$$

To obtain $\lambda^{\phi}_{k,j}$, we minimize the cost function with respect $\lambda^{\phi}_{k,j}$ and set it to zero which results:

$$\lambda^{\phi}_{k,j} = \frac{1}{H^{\phi}_{k,j}} \tag{46}$$

provided that $H^{\phi}_{k,j} \neq 0$. However, it has been observed in many cases that optimizing the factor matrices with β-divergence and the sparseness in Equation (46) increases the likelihood for some $H^{\phi}_{k,j}$ to converge very close to zero, thus leading to numerical divergence when dividing by zero. Other practices introduced a small constant to $H^{\phi}_{k,j}$ to prevent direct division by zero. Unfortunately, such approach is identical to constant sparsity and no longer preserves the L_1-norm optimal solution. In this paper, we adopt the maximum likelihood approach to formulating the adaptive estimation of sparsity parameter $\lambda^{\phi}_{k,j}$. Considering the following maximum likelihood criterion [31,36]:

$$\boldsymbol{\lambda}^{ML} = \arg\max_{\boldsymbol{\lambda}} \ln p(\mathbf{v}|\boldsymbol{\lambda}, \check{\mathbf{W}}) \tag{47}$$

where $\ln p(\mathbf{v}|\boldsymbol{\lambda}, \check{\mathbf{W}})$ is the log-likelihood conditional probability of the observations given $\check{\mathbf{W}}$ and $\boldsymbol{\lambda}$. By using the Jensen's inequality, for any distribution $Q(\mathbf{h})$, the log-likelihood function satisfies the following:

$$\begin{aligned} \boldsymbol{\lambda}^{ML} &= \arg\max_{\boldsymbol{\lambda}} \textstyle\int Q(\mathbf{h}) \ln p(\mathbf{v}, \mathbf{h}|\boldsymbol{\lambda}, \check{\mathbf{W}}) d\mathbf{h} \\ &= \arg\max_{\boldsymbol{\lambda}} \textstyle\int Q(\mathbf{h}) (\ln p(\mathbf{v}|\mathbf{h}, \check{\mathbf{W}}) + \ln p(\mathbf{h}|\boldsymbol{\lambda})) d\mathbf{h} \\ &= \arg\max_{\boldsymbol{\lambda}} \textstyle\int Q(\mathbf{h}) \ln p(\mathbf{h}|\boldsymbol{\lambda}) d\mathbf{h} \end{aligned} \tag{48}$$

Since each element of $\mathbf{H}$ is constrained to be exponential distributed with independent decay parameters, Equation (48) becomes:

$$\lambda^{ML}_{g} = \arg\max_{\lambda} \int Q(\mathbf{h}) \left(\ln \lambda_g - \lambda_g h_g\right) d\mathbf{h} \tag{49}$$

Thus, we have

$$\lambda^{ML}_{g} = \frac{1}{\int h_g Q(\mathbf{h}) d\mathbf{h}} \tag{50}$$

One can easily check that the distribution that maximizes the maximum likelihood is given by $Q(\mathbf{h}) = p(\mathbf{h}|\mathbf{v}, \boldsymbol{\lambda}, \check{\mathbf{W}}) = p(\mathbf{v}|\mathbf{h}, \boldsymbol{\lambda}, \check{\mathbf{W}})p(\mathbf{h}|\boldsymbol{\lambda})/p(\mathbf{v}|\boldsymbol{\lambda}, \check{\mathbf{W}})$ which is the posterior distribution of $\mathbf{h}$ and $p(\mathbf{v}|\mathbf{h}, \boldsymbol{\lambda}, \check{\mathbf{W}})$ is the log-likelihood function of the observation which is usually expressed by a Gaussian density function with mean centered at $\sum\limits_{k,\tau,\phi} \overline{W}^{\tau}_{i-\phi,k} H^{\phi}_{k,j-\tau}$. However, as $H^{\phi}_{k,j}$ is directly acquired from the original code matrix $H^{0}_{k,j}$, we can simply work with $\tau_{max} = 0$. This allows us to express the log-likelihood function of the posterior distribution of $\mathbf{h}$ up to a constant as

$$\begin{aligned} \ln p(\mathbf{h}|\mathbf{v}, \boldsymbol{\lambda}, \check{\mathbf{W}}) &\doteq \ln p(\mathbf{v}|\mathbf{h}, \boldsymbol{\lambda}, \check{\mathbf{W}}) + \ln p(\mathbf{h}|\boldsymbol{\lambda}) \\ &\doteq \tfrac{1}{2}||vec(\mathbf{V}) - \sum_{\phi}\left(\mathbf{I} \otimes \overset{\downarrow\phi}{\overline{\mathbf{W}}}\right) vec\left(\mathbf{H}^{\phi}\right)||_F^2 + \alpha \sum_{\phi}\left\{ \left(\boldsymbol{\lambda}^{\phi}\right)^T vec\left(\mathbf{H}^{\phi}\right) - \left(\log \boldsymbol{\lambda}^{\phi}\right)^T 1\right\} \\ &= F(\mathbf{H}, \boldsymbol{\lambda}) \end{aligned} \tag{51}$$

where "$\doteq$" denotes equality up to a constant, "$\otimes$" is the Kronecker product, $\mathbf{1}$ is vector contains unit elements, $\mathbf{I}$ is the identity matrix, α assumes the role of a regularization constant to balance the cost function fit and smoothness of $\mathbf{H}$. For ease of presentation, we simplify the above terms as $\mathbf{v} = vec(\mathbf{V})$, $\check{\mathbf{W}} = \left[\begin{array}{ccc} \mathbf{I} \otimes \overset{\downarrow 0}{\overline{\mathbf{W}}} & \ldots & \mathbf{I} \otimes \overset{\downarrow \phi_{max}}{\overline{\mathbf{W}}} \end{array}\right]$, $\mathbf{h} = \{h_g\} = \left[\begin{array}{ccc} vec\left(\mathbf{H}^0\right)^T & \cdots & vec\left(\mathbf{H}^{\phi_{max}}\right)^T \end{array}\right]^T$, $\boldsymbol{\lambda} = \{\lambda_g\} = \left[\begin{array}{ccc} \boldsymbol{\lambda}^{0T} & \ldots & \boldsymbol{\lambda}^{\phi_{max}T} \end{array}\right]^T$ which enables us to rewrite Equation (46) as

$$F(\mathbf{H}, \boldsymbol{\lambda}) = \frac{1}{2}||\mathbf{v} - \check{\mathbf{W}}\mathbf{h}||_F^2 + \alpha\left(\boldsymbol{\lambda}^T\mathbf{h} - (\log\boldsymbol{\lambda})^T\mathbf{1}\right) \tag{52}$$

For ease of analysis, $Q(\mathbf{h})$ is represented using Gibbs distribution as $Q(\mathbf{h}) = \frac{1}{Z}\exp(-F(\mathbf{h}))$ where $Z = \int \exp(-F(\mathbf{h}))d\mathbf{h}$. Let P represents the index set of inactive code i.e., $P = \left\{\phi, k, j | H^{\phi}_{k,j} = 0\right\}$ and M the index set of active code i.e., $M = \left\{\phi, k, j | H^{\phi}_{k,j} \neq 0\right\}$. Thus, $Q(\mathbf{h})$ can be factorized as

$$\begin{aligned} Q(\mathbf{h}) &= \tfrac{1}{Z}\exp(-F(\mathbf{h}, \boldsymbol{\lambda})) \\ &\approx \tfrac{1}{Z_P}\exp(-F(\mathbf{h}_P, \boldsymbol{\lambda}_P))\tfrac{1}{Z_M}\exp(-F(\mathbf{h}_M, \boldsymbol{\lambda}_M)) \\ &= Q_P(\mathbf{h}_P)Q_M(\mathbf{h}_M) \end{aligned} \tag{53}$$

Since $\mathbf{h}_M$ corresponds to the original non-zero value of $\mathbf{h}$, it then follows that $Q_M(\mathbf{h}_M)$ is not of interest to us. We are only interested in $\mathbf{h}_P$ and therefore, to characterize $Q_P(\mathbf{h}_P)$, we need to allow some positive deviation to $\mathbf{h}_P$. A suitable distribution is to use the factorized exponential distribution given by

$$\hat{Q}_P(\mathbf{h}_P \geq 0) = \prod_{p \in P} \frac{1}{u_p}\exp\left(-\frac{h_p}{u_p}\right) \tag{54}$$

as the approximate distribution. The variational parameters $u = \{u_p\}$ are determined by minimizing the Kullback–Leibler divergence between true Q_P and approximate $\hat{Q}_P$:

$$\begin{aligned} \boldsymbol{u} &= \arg\min_{\boldsymbol{u}} \int \hat{Q}_P(\mathbf{h}_P)\ln\tfrac{\hat{Q}_P(\mathbf{h}_P)}{Q_P(\mathbf{h}_P)}d\mathbf{h}_P \\ &= \arg\min_{\boldsymbol{u}} \int \hat{Q}_P(\mathbf{h}_P)\{\ln\hat{Q}_P(\mathbf{h}_P) - Q_P(\mathbf{h}_P)\}d\mathbf{h}_P \end{aligned} \tag{55}$$

which leads to the following optimization:

$$\min_{u_p} \boldsymbol{b}_P^T\boldsymbol{u} + \frac{1}{2}\, u^T\mathbf{C}\boldsymbol{u} - \sum_{p \in P}\ln u_p \tag{56}$$

where $\boldsymbol{b}_P = (\mathbf{Ch} - \check{\mathbf{W}}^T\mathbf{v} + \boldsymbol{\lambda})_P$ and $\mathbf{C} = \mathbf{C}_P + diag(\mathbf{C}_P)$ with $\mathbf{C} = \check{\mathbf{W}}^T\check{\mathbf{W}}$, $\mathbf{C}_P = \check{\mathbf{W}}_P^T\check{\mathbf{W}}_P$. Solving Equation (56) for u_p leads to the following update:

$$u_p \leftarrow u_p \frac{-b_p + \sqrt{b_p^2 + 4\frac{(\mathbf{C}\boldsymbol{u})_p}{u_p}}}{2(\mathbf{C}\boldsymbol{u})_p} \tag{57}$$

Once u_p is obtained and re-arranged to the original form $u_{k,j}^{\phi}$, the final update for $\lambda_{k,j}^{\phi}$ takes the form of:

$$\lambda_{k,j}^{\phi} = \frac{1}{H_{k,j}^{\phi} + \delta_{k,j}^{\phi}} \tag{58}$$

where

$$\delta_{k,j}^{\phi} = \begin{cases} 0 \text{ if } H_{k,j}^{\phi} \neq 0 \\ u_{k,j}^{\phi} \text{ if } H_{k,j}^{\phi} = 0 \end{cases} \tag{59}$$

Equipped with above, we obtain the multiplicative update for the normalized $\mathbf{W}$ as

$$\mathbf{W}^{\tau} = \overline{\mathbf{W}}^{\tau} \cdot \left[\frac{\sum_{\phi} \left(\overset{\uparrow\phi}{\mathbf{V}} \cdot \overset{\uparrow\phi}{\overline{\boldsymbol{\Lambda}}}^{(\beta-2)} \right) \overset{\rightarrow\tau}{\mathbf{H}^{\phi}}^{T} + \overline{\mathbf{W}}^{\tau} diag\left(\sum_{\tau} 1 \left(\left(\overset{\uparrow\phi}{\overline{\boldsymbol{\Lambda}}}^{(\beta-1)} \overset{\rightarrow\tau}{\mathbf{H}^{\phi}}^{T} \right) \cdot \overline{\mathbf{W}}^{\tau} \right) \right)}{\sum_{\phi} \overset{\uparrow\phi}{\overline{\boldsymbol{\Lambda}}}^{(\beta-1)} \overset{\rightarrow\tau}{\mathbf{H}^{\phi}}^{T} + \overline{\mathbf{W}}^{\tau} diag\left(\sum_{\tau} 1 \left(\left(\left(\overset{\uparrow\phi}{\mathbf{V}} \cdot \overset{\uparrow\phi}{\overline{\boldsymbol{\Lambda}}}^{(\beta-2)} \right) \overset{\rightarrow\tau}{\mathbf{H}^{\phi}}^{T} \right) \cdot \overline{\mathbf{W}}^{\tau} \right) \right)} \right]^{-\delta(\beta)} \tag{60}$$

for $\tau = 0,\ 1,\ \ldots,\ \tau_{max}$. By using the same approach, we can obtain the update for the sparse $\mathbf{H}$ as follows:

$$\mathbf{H}^{\phi} = \mathbf{H}^{\phi} \cdot \left[\frac{\sum_{\tau} \overset{\downarrow\phi}{\overline{\mathbf{W}}^{\tau}}^{T} \left(\overset{\leftarrow\tau}{\mathbf{V}} \cdot \overset{\leftarrow\tau}{\overline{\boldsymbol{\Lambda}}}^{(\beta-2)} \right)}{\sum_{\tau} \overset{\downarrow\phi}{\overline{\mathbf{W}}^{\tau}}^{T} \overset{\leftarrow\tau}{\overline{\boldsymbol{\Lambda}}}^{(\beta-1)} + \alpha\boldsymbol{\lambda}^{\phi}} \right]^{\delta(\beta)} \tag{61}$$

for $\phi = 0,\ 1,\ \ldots,\ \phi_{max}$. In Equation (61), α assumes the role of a regularization constant to balance the cost function fit and smoothness of $\mathbf{H}$. In this work, we set $\alpha \in [0.5,\ 1]$ which has been found to give satisfactory results.

2.5. Optimizing the Fractional β

To determine the optimal value for β, we perform the investigation from the source separation viewpoint. Mathematically, the single-channel signal separation (SCSS) [37–39] problem can be treated as one mixture of N unknown source signals:

$$y(t) = x_1(t) + x_2(t) + \ldots + x_N(t) \tag{62}$$

where $t = 1,\ 2,\ \ldots,\ T$ denotes time index and the goal is to estimate the sources $x_k(t)$, $\forall k \in N$ of length T when only the observation signal $y(t)$ is available. For simplicity, we consider only $N = 2$ sources in the mixture. We also use 50 different pieces of piano music, 50 different pieces of trumpet music and 50 different pieces of violin music from the RWC [40] database to generate different mixtures. The signal-to-distortion (SDR) [41] is used to measure the performance. The SDR results and its corresponding β value that produces the best performance in the separation of mixtures are shown in Table 2. From these experiments, we can propose some general ideas of how to choose a suitable β value: (i) The mixtures from same type of music share similar β value, e.g., the best results of piano and trumpet mixture occur around β = 2 but the best results of piano and violin mixture occur around $\beta = 1$. (ii) If the power of one source is clearly weaker than the other source, then a smaller β value should be selected. (iii) When there is a large amount of overlap between the two sources in the in the time–frequency domain, a larger β value should be selected.

Table 2. Results using different sources.

Mixtures	SDR (dB)	β
Piano + trumpet	16.11	2.11
	9.19	2.13
	9.43	1.93
	7.73	1.82
	12.21	2.09
Piano + violin	13.07	1.07
	8.15	1.23
	6.25	0.92
	9.33	1.20
	8.19	0.89
Trumpet + violin	14.63	0.68
	8.14	0.62
	7.81	0.67
	9.81	0.51
	7.55	0.52

Table 2 strongly suggests that the β value depends on the mixture of original sources. Generally, it depends mainly on the two factors: (i) the weight of each source in the mixture; and (ii) the frequency spread of each source which the frequency band contains most weight of the signal. Firstly, we define weight of each source in the mixture by the following function:

$$\gamma_k = 1 - \frac{|x_k(t) - y(t)|}{\sum_{l=1}^{N} |x_l(t) - y(t)|^2} \tag{63}$$

for $k = 1, \ldots, N$. The term γ_k is nonnegative and bounded to unity. It measures the dominance of k-th source in the mixture. The higher value of γ_k, the greater the contribution from the k-th source is to the mixture. Secondly, we consider the separability of each source in the time–frequency domain by the following function:

$$\eta_k = \frac{||M_k(i,j) X_k(i,j)||_F^2 - ||M_k(i,j) \sum_{l=1, l \neq k}^{N} X_l(i,j)||_F^2}{||X_k(i,j)||_F^2} \tag{64}$$

where $||\cdot||_F$ is the Frobenius norm, $X_k(i,j)$ is the short-time Fourier Transform (STFT) of $x_k(t)$ with i representing the frequency bins and j the timeslot, and $M_k(i,j)$ is the binary mask Obtained from the k-th source as

$$M_k(i,j) = \begin{cases} 1 & \text{if} |X_k(i,j)|^2 > |X_l(i,j)|^2 \\ 0 & \text{otherwise} \end{cases} \tag{65}$$

The function η_k is also nonnegative and determines the degree separability of the signal in each frequency band. Based on the experiments conducted, both γ_k and η_k have an inverse relation to β. Thus, one possible empirical approach to determine β is proposed as follows:

$$\beta(n+1) = \rho(n)\beta(n) + (1 - \rho(n))\min\left[\left(\sum_{k=1}^{N} \frac{\varepsilon_1 \cdot \eta_k + (1 - \varepsilon_1) \cdot \gamma_k}{\gamma_k \eta_k}\right), \varepsilon_2\right] \tag{66}$$

where $\rho(n)$ is step size, ε_1 is a constant to weight the effects of η_k and γ_k. For example, in the experiments conducted, we have given more emphasis to γ_k and set $\varepsilon_1 = 1/3$. The term ε_2 is a constant to control the value of $\beta(n+1)$ to ensure its value is bounded within an interval chosen by the user (for example, in the experiments conducted we have set $\varepsilon_2 = 4$ as normally does not exceed 4). Equation (66) is inserted into the update funtions in Equations (60) and (61) to update β at every iteration in conjunction with the update of **W** and **H**. In this case, β can be optimized based on the type of sources and the separation process. This enables the separation process to be fully automated and enables more accurate performance. In the case of SCSS, the sources are unknown and these are estimated from the mixture as:

$$\hat{X}_k(i,j) = M_k(i,j) Y(i,j) \tag{67}$$

where

$$M_k(i,j) = \begin{cases} 1\ if\ |\check{X}_k(i,j)|^2 > |\check{X}_l(i,j)|^2 \\ 0\ otherwise \end{cases} \tag{68}$$

and

$$|\check{X}_k(i,j)|^2 = \sum_{\tau=0}^{\tau_{max}} \sum_{\phi=0}^{\phi_{max}} \overline{W}_{i-\phi,k}^{\tau} H_{k,j-\tau}^{\phi} \tag{69}$$

The expression in (69) is computed using the time–frequency deconvolution model. The main steps of the proposed algorithm have been summarized in Algorithm 1.

Algorithm 1. Overview Proposed Algorithm

```
1.  Initialize $\mathbf{W}^{\tau}$ and $\mathbf{H}^{\phi}$ with non-negative random values.
2.  Compute the STFT:
        $Y(i,j) = \text{STFT}(y(t))$, and let $V_{i,j} = |Y(i,j)|^2$.
3.  Compute $\overline{\Lambda}_{i,j} = \sum_{k,\tau,\phi} \overline{W}_{i-\phi,k}^{\tau}\, H_{k,j-\tau}^{\phi}$.
4.  Compute $u_p \leftarrow u_p \left\{ \left( -b_p + \sqrt{b_p^2 + 4\frac{(\mathbf{C}\boldsymbol{u})_p}{u_p}} \right) / 2(\mathbf{C}\boldsymbol{u})_p \right\}$
5.  Assign $\lambda_{k,j}^{\phi} = \frac{1}{H_{k,j}^{\phi}+\delta_{k,j}^{\phi}}$ where $\delta_{k,j}^{\phi} = \begin{cases} 0 \text{ if } H_{k,j}^{\phi} \neq 0 \\ u_{k,j}^{\phi} \text{ if } H_{k,j}^{\phi} = 0 \end{cases}$
6.  Update $\mathbf{H}^{\phi} = \mathbf{H}^{\phi}\cdot \left[ \frac{\sum_\tau \overset{\downarrow\phi}{\overline{\mathbf{W}}^{\tau}}{}^{T} \left( \overset{\leftarrow\tau}{\mathbf{V}} \cdot \overset{\leftarrow\tau}{\overline{\mathbf{\Lambda}}}{}^{(\beta-2)} \right)}{\sum_\tau \overset{\downarrow\phi}{\overline{\mathbf{W}}^{\tau}}{}^{T}\ \overset{\leftarrow\tau}{\overline{\mathbf{\Lambda}}}{}^{(\beta-1)} + \alpha\boldsymbol{\lambda}^{\phi}} \right]^{\delta(\beta)}$
7.  Compute $\overline{\Lambda}_{i,j} = \sum_{k,\tau,\phi} \overline{W}_{i-\phi,k}^{\tau}\, H_{k,j-\tau}^{\phi}$.
8.  Update the spectral bases:
        $\mathbf{W}^{\tau} = \overline{\mathbf{W}}^{\tau}\cdot \left[ \frac{\sum_\phi \left( \overset{\uparrow\phi}{\mathbf{V}} \cdot \overset{\uparrow\phi}{\overline{\mathbf{\Lambda}}}{}^{(\beta-2)} \right) \overset{\rightarrow\tau}{\mathbf{H}^{\phi}}{}^{T} + \overline{\mathbf{W}}^{\tau} diag \left( \sum_\tau 1 \left( \left( \overset{\uparrow\phi}{\overline{\mathbf{\Lambda}}}{}^{(\beta-1)} \overset{\rightarrow\tau}{\mathbf{H}^{\phi}}{}^{T} \right) \cdot \overline{\mathbf{W}}^{\tau} \right) \right)}{\sum_\phi \overset{\uparrow\phi}{\overline{\mathbf{\Lambda}}}{}^{(\beta-1)} \overset{\rightarrow\tau}{\mathbf{H}^{\phi}}{}^{T} + \overline{\mathbf{W}}^{\tau} diag \left( \sum_\tau 1 \left( \left( \left( \overset{\uparrow\phi}{\mathbf{V}} \cdot \overset{\uparrow\phi}{\overline{\mathbf{\Lambda}}}{}^{(\beta-2)} \right) \overset{\rightarrow\tau}{\mathbf{H}^{\phi}}{}^{T} \right) \cdot \overline{\mathbf{W}}^{\tau} \right) \right)} \right]$
9.  For $k = 1, \ldots, N$, compute:
        $|\check{X}_k(i,j)|^2 = \sum_{\tau=0}^{\tau_{max}} \sum_{\phi=0}^{\phi_{max}} \overline{W}_{i-\phi,k}^{\tau} H_{k,j-\tau}^{\phi}$
        $M_k(i,j) = \begin{cases} 1\ if\ |\check{X}_k(i,j)|^2 > |\check{X}_l(i,j)|^2 \\ 0\ otherwise \end{cases}$
        $\hat{X}_k(i,j) = M_k(i,j) Y(i,j)$
        $\hat{x}_k(t) = \text{STFT}^{-1}\left[\hat{X}_k(i,j)\right]$
        $\gamma_k = 1 - \frac{|\hat{x}_k(t)-y(t)|}{\sum_{l=1}^{N} |\hat{x}_l(t)-y(t)|^2}$
        $\eta_k = \frac{M_k(f,t)\hat{X}_k(f,t)_F^2 - M_k(f,t)\sum_{l=1,l\neq k}^{N} \hat{X}_l(f,t)_F^2}{\hat{X}_k(f,t)_F^2}$
        $\beta \leftarrow \rho\beta + (1-\rho)\min\left[ \left( \sum_{k=1}^{N} \frac{\varepsilon_1\eta_k + (1-\varepsilon_1)\gamma_k}{\gamma_k\eta_k} \right),\ \varepsilon_2 \right]$
10. Repeat Steps 3–9 until it converges or reaches the pre-defined number of iteration.
```

3. Experiments, Results and Analysis

In this section, we conduct in-depth investigations of the proposed algorithm to analyze the impact of fixed and adaptive sparsity, the adaptive behavior of the sparsity parameter, and the analysis of fractional β-divergence. The analysis is necessary as the issue of sparsity of the temporal codes would undermine the quality of matrix factorization when the β value is inappropriately chosen. The selection of the β value should consider the sparseness constraint used in the cost function. In addition, the proposed algorithm based on matrix factor time–frequency deconvolution is also compared to conventional NMF models. This allows us to quantify the impacts of fractional β-divergence and sparsity behaviors when using the time–frequency deconvolution model.

3.1. Experimental Set-Up

To investigate the proposed method, we use the algorithm to separate several pieces of mixed music signals. Several experimental simulations under different conditions have been designed to investigate the efficacy of the proposed method. All simulations were performed using MATLAB as the programming platform and performed using a PC with dual core processor @ 2.4 GHz (i7 Intel processor) 8 GB RAM and 320 GB HDD. The tested signals are generated by mixing several music sources. The polyphonic music is 4 s long and the sampling frequency is 16 kHz. In this experiment, we randomly chose 50 different pieces of piano music, 50 different pieces of trumpet music and 50 different pieces of violin music from the RWC database to produce the different mixtures. The mixed signal was then generated by adding the chosen sources. In all cases, the sources were mixed with equal average power over the duration of the signals. The time–frequency (TF) representation was obtained by first normalizing the time-domain signal to unit power and then by computing the STFT using 2048 point Hanning window FFT with a 50% overlap. We evaluated our separation performance in terms of the signal-to-distortion ratio (SDR) which is one form of perceptual measure. This is a global measure that unifies source-to-interference ratio (SIR), source-to-artifacts ratio (SAR) and source-to-noise ratio (SNR). The definition and mathematical expression and MATLAB routines for computing these criteria can be obtained online [42].

3.2. Analysis of Adaptive and Fixed Sparsity

In this implementation, we conducted several experiments to compare the performance of the proposed method using different β values. Our aim was to investigate the impact of β value used in the separation. Figure 1 shows the time and TF domains of the original trumpet, piano music and its mixture. The TF domain is displayed using the log-frequency spectrogram. The trumpet and the piano play a different short melodic passage each consisting of three distinct notes. However, both trumpet and piano overlap in time, and the piano notes are interspersed in frequency with the trumpet notes. Hence, this is a challenging task for single channel separation which tests the impact of flexible β for matrix factorization.

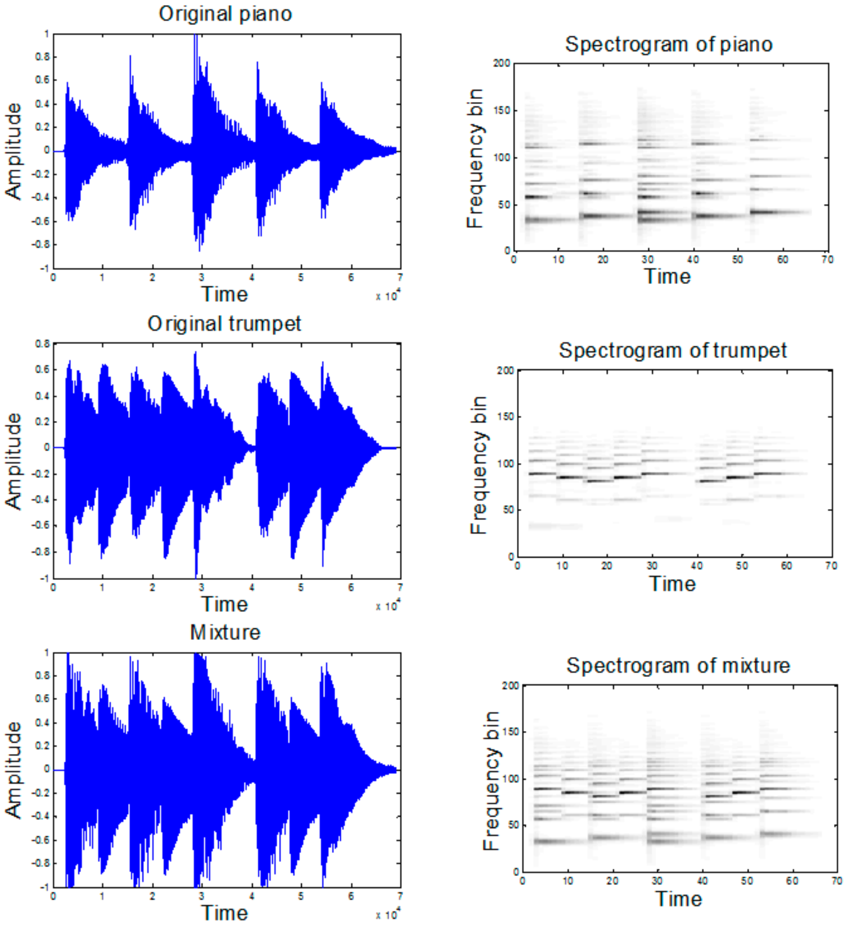
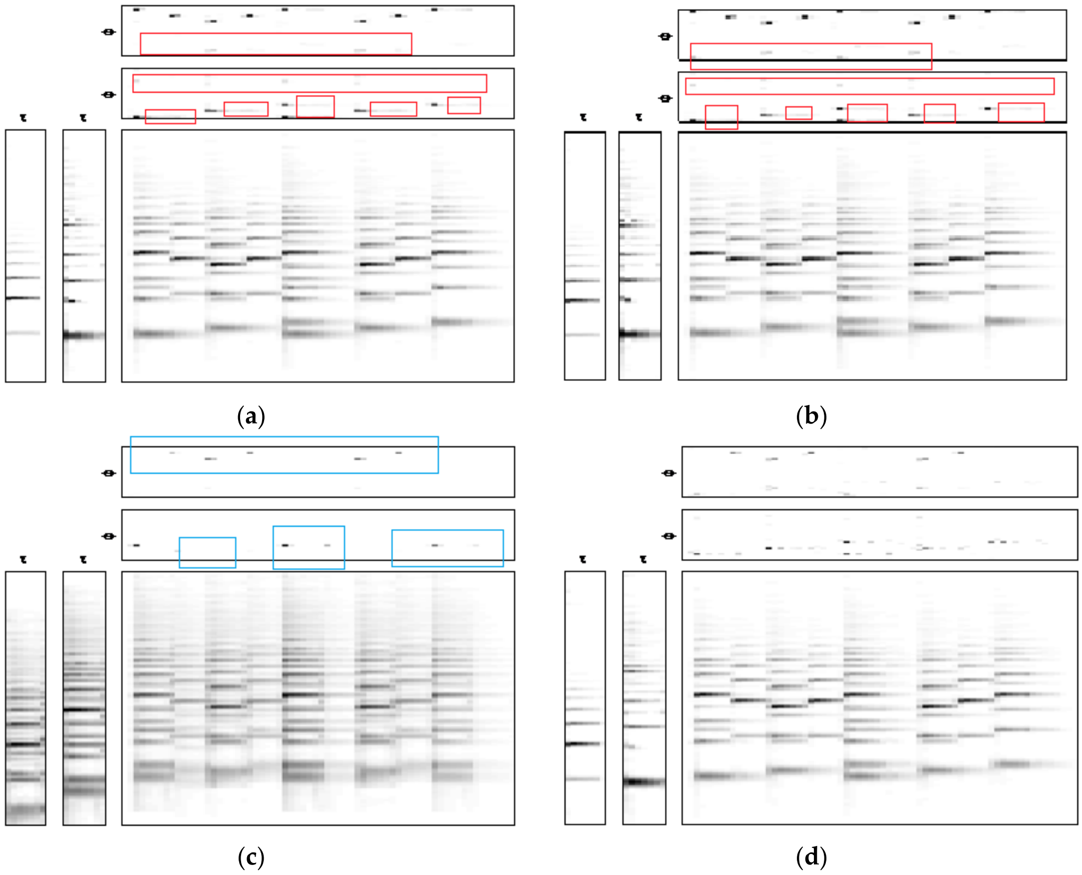


Figure 1. Time-domain representation and log-frequency spectrogram of: the piano music (**top panels**); the trumpet music (**middle panels**); and the mixed signal (**bottom panels**).

Figure 2 shows the estimation of bases $\mathbf{W}^{\tau}$ and temporal codes $\mathbf{H}^{\phi}$ when using different λ values. In Figure 2a, we set $\lambda = 0$ which renders non-sparse solution. There is obvious spreading of the estimated temporal codes, as shown in the red part of the figure. In Figure 2b, when $\lambda = 0.1$, there are some improvements over the spreading but they still exist in the red parts. Here, the sparseness is not strong enough and, as a result, the estimated mixture becomes under-sparse. In Figure 2c, when $\lambda = 100$, it is visibly shown in the blue parts that some information has been lost in the estimated temporal codes and the resulting estimated mixture becomes very noisy. Finally, Figure 2d shows the case where the sparseness parameters are adaptively and individually estimated using the prior information of $\mathbf{H}$. The obtained result has shown that the estimated temporal codes are just appropriately sparse and, by visual inspection, the resulting estimated mixture retains all information, as evidenced by the musical notes, while the noise level has been kept small, which very closely resembles the original mixture.



(**a**) (**b**) (**c**) (**d**)

Figure 2. Estimation of $\mathbf{W}^{\tau}$ and $\mathbf{H}^{\phi}$ with: (**a**) $\lambda = 0$; (**b**) $\lambda = 0.1$; (**c**) $\lambda = 100$; (**d**) $\lambda = adaptive$.

3.3. Adaptive Behavior of Sparsity Parameter

In this section, we use the piano and trumpet mixture, and fix β value to 1 and the λ value ranges from 0 to 3 with each step of 0.1 to show the results. The adaptive behavior of the sparsity parameters using the proposed method is demonstrated. Figure 3 presents the convergence trajectory of four adaptive sparsity parameters, $\lambda_{1,1}^{\phi=0}$, $\lambda_{1,5}^{\phi=0}$, $\lambda_{1,10}^{\phi=0}$ and $\lambda_{1,15}^{\phi=0}$, corresponding to their respective element codes, $h_{1,1}^{\phi=0}$, $h_{1,5}^{\phi=0}$, $h_{1,10}^{\phi=0}$ and $h_{1,15}^{\phi=0}$. All sparsity parameters are initialized as $\lambda = 1$. After 150 iterations, the above sparsity parameters converge to their steady-states. By examining Figure 3, it is noted that the converged steady-state values are significantly different for each sparsity parameter, e.g., $\lambda_{1,1}^{\phi=0} = 0.9$, $\lambda_{1,5}^{\phi=0} = 0.18$, $\lambda_{1,10}^{\phi=0} = 0.29$ and $\lambda_{1,15}^{\phi=0} = 0.08$, even though they started at the same initial condition. This shows that each element code has its own sparseness.

In Figure 4, we compare the SDR results of using different λ values. In the figure, we can see the λ value that can get the best result changes with the mixture. For each different mixture, the best λ values are different. In Figure 4, we can see the best separation results of piano and trumpet mixture occurs near $\lambda = 1$, and the SDR = 14.7 dB. However, as λ increases, the SDR performance begins to deteriorate rapidly due to over-sparseness of the temporal code $\mathbf{H}^{\phi}$.

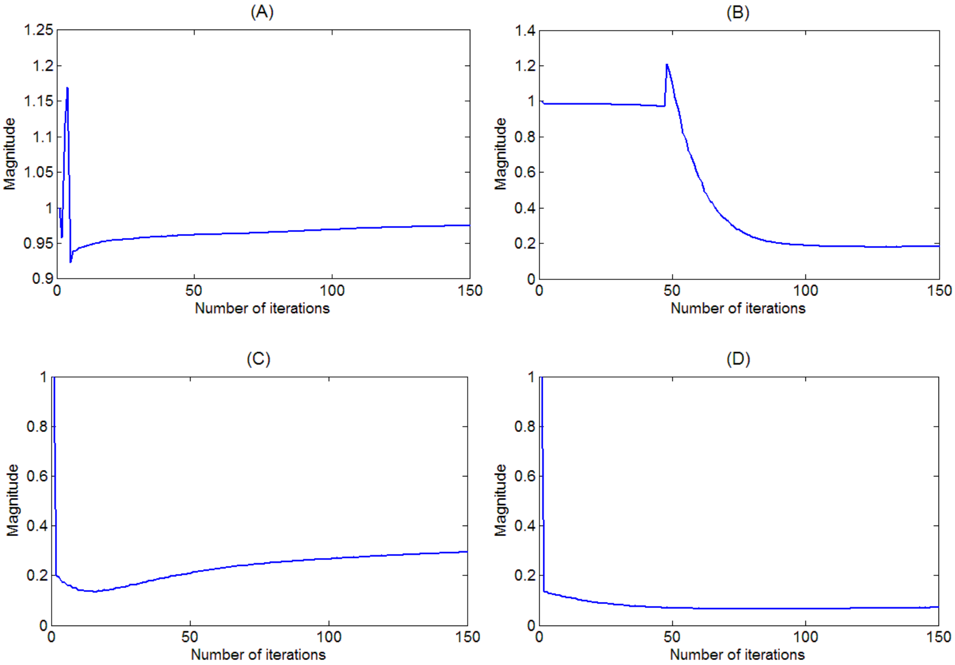


Figure 3. Convergence trajectory of the sparsity parameter: (**A**) $\lambda_{1,1}^{\phi=0}$; (**B**) $\lambda_{1,5}^{\phi=0}$; (**C**) $\lambda_{1,10}^{\phi=0}$; and (**D**) $\lambda_{1,15}^{\phi=0}$.

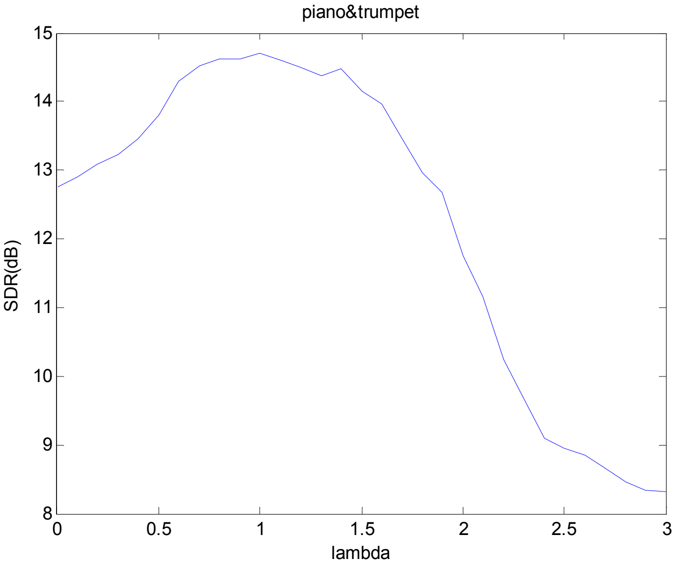


Figure 4. SDR results of piano and trumpet mixture when using different λ values.

Although the SDR performance in Figure 4 seems to suggest good performance, it may not necessary refer to the optimum setting. In fact, when λ is fixed to a constant value, the matrix factors deconvolution process may still be subjected to under- and over-fitting. In the previous sparsity algorithm, λ is fixed for the whole process and this sparsity may not necessarily suited for the whole signal. This calls for the need to allow each temporal code to have its own sparsity parameter. In the adaptive sparsity algorithm in Equation (53), the sparsity parameter λ is updated alongside $\mathbf{W}^{\tau}$ and $\mathbf{H}^{\phi}$ in the process. Therefore, the sparsity parameter is optimized for each element of the temporal code. In addition, we plot the histogram of the converged adaptive sparsity parameters in Figure 5. The plot strongly suggests that the histogram can be represented as a bimodal distribution. We have used the Gaussian mixture model (GMM) [43] to learn the distribution of this histogram and the result produces two Gaussian distributions with mean 0.16 and 1.1. The global mean of the GMM is given by 0.92.

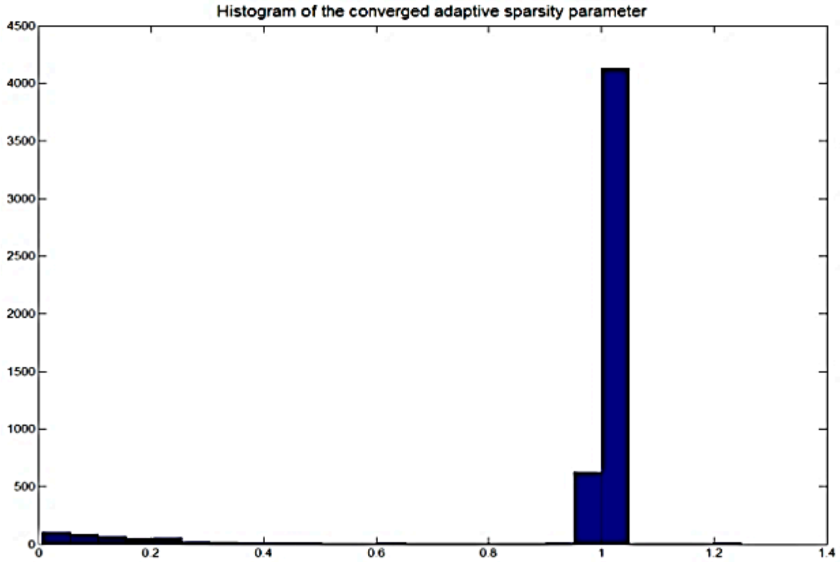


Figure 5. Histogram of the converged adaptive sparsity parameter.

With the GMM analysis, we can proceed to further investigate the assignment of sparsity parameters and compare them with the adaptive approach. We considered the following sparseness assignments:

Case (1): No sparseness $\lambda = 0$.
Case (2): Uniform and constant sparseness $\lambda = 0.16$ corresponding to the mean of the first Gaussian distribution of the GMM.
Case (3): Uniform and constant sparseness $\lambda = 1.1$ corresponding to the mean of the second Gaussian distribution of the GMM.
Case (4): Uniform and constant sparseness $\lambda = 0.92$ corresponding to the global mean of the converged adaptive sparsity.
Case (5): Uniform and constant sparseness $\lambda = 1$ corresponding to the global mean of the converged adaptive sparsity.
Case (6): Maximum likelihood adaptive sparseness, i.e., Equation (55).

The SDR results are tabulated in Table 3 where we can see the separation results of all the six cases. The obtained results readily informed that the source separation with adaptive sparsity has rendered the best separation result.

Table 3. Results of separation for different mixture.

Methods	SDR (dB)
Case (1)	12.77
Case (2)	13.01
Case (3)	14.60
Case (4)	14.62
Case (5)	14.70
Case (6)	15.60

3.4. Analysis of Fractional β-Divergence

Figure 6 shows the SDR values of the separation results using different values of β values. In this implementation, we fixed the sparsity parameter λ to 0, and the value of β ranges from 0 to 4, and each step is 0.1. In the figure, we can observe that the SDR value changes as the β value is increased. The best result is obtained at when $\beta = 2.5$, where SDR = 14.26 dB. The SDR value keeps increasing for values of β value within the range of $0 \leq \beta < 2.5$, and, after the best performance is attained, the performance deteriorates as β increases. In this figure, we can see that the best performance does not necessarily occur at value of β used other algorithms, i.e., $\beta = 2,\ 1,\ 0$. This means, if we choose the best β to carry out the separation, we can obtain better results than the other algorithms.

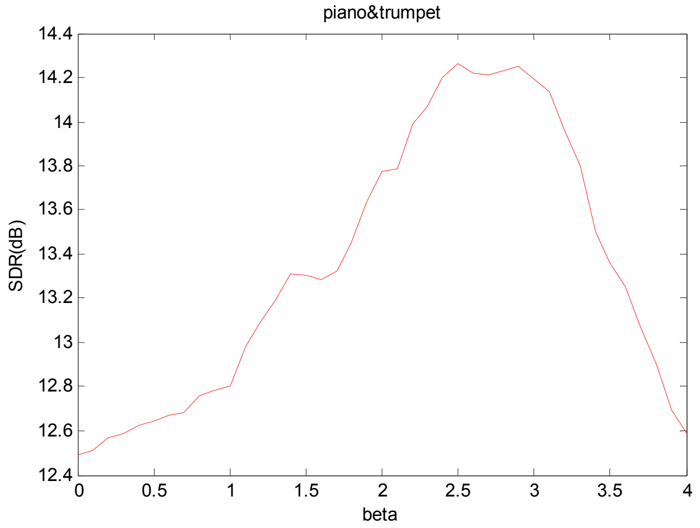


Figure 6. SDR values of the separation results of mixture using different β values.

We also conducted several experiments to compare the performance of the proposed method with the non-sparse and normal sparse methods using different β value. To investigate the impact of β and λ value used in the separation, we used β that ranges from 0 to 4 (with every increment of 0.1), and λ was set to several configurations, i.e., non-sparse, best fixed sparsity that is obtained in Figure 4 and the proposed adaptive sparsity. The results are plotted in Figure 7, which shows the mutual influence between λ and λ. It is noted that the SDR performance of non-sparse algorithm is the lowest of which its best result occurs when $\beta = 2.5$ with SDR = 14.26 dB. The normal sparse algorithm using best λ value gives better performance than the non-sparse method with its best result occuring when $\beta = 0.5$ with SDR = 15.96 dB. On the other hand, the proposed algorithm with adaptive λ delivers the best performance where β occurs around 0.3 with SDR = 16.71 dB. It is also noted that the worst SDR performance given by $\beta = 2.9$ with adaptive λ is still higher than the highest SDR when $\beta = 2.5$ without sparsity optimization $\lambda = 0$.

All the above experiments used the pre-determined β that are fixed for the whole process. In this section, we present the results of experiments where β is adaptively tuned alongside with the adaptation of $\mathbf{W}^{\tau}$, $\mathbf{H}^{\phi}$ and $\boldsymbol{\lambda}^{\phi}$. We applied this method to the source separation problem and compared the results with the situation where $\beta = 1$ and $\beta = 2$ corresponding to the Kullback–Leibler divergence and Least Square cost function, respectively. In the update of adaptive β, the step size $\rho(n) = 0.95^n$ is selected which represents an exponential decay update process. The SDR results of the top five best performance are tabulated in Table 4. It is interesting to note from the table that the proposed algorithm with adaptive β delivers better performance by about 2.1 dB compared to that when $\beta = 1$ (Kullback–Leibler divergence) and 1.9 dB compared to that when $\beta = 2$ (Least Square distance). The obtained performance improvement is attributed to the fact that the joint optimization of β with $\mathbf{W}^{\tau}$, $\mathbf{H}^{\phi}$ and λ^{ϕ} has enabled the current estimate of β to better fit mixture of sources and thus rendered better source separation performance.

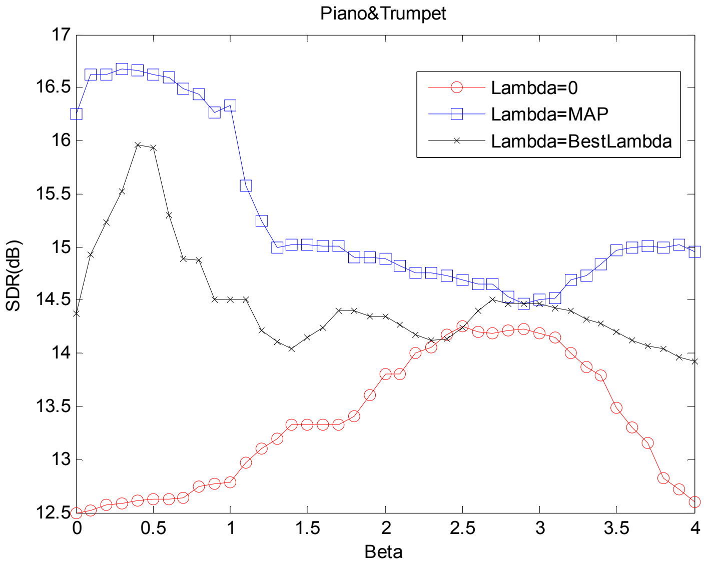


Figure 7. SDR values of the separation results of mixture using different β values and sparse methods.

Table 4. SDR (dB) results of adaptive β versus fixed β.

Mixtures	SDR (dB) Using Adaptive β	SDR (dB) Using $\beta = 1$	SDR (dB) Using $\beta = 2$
Piano + Trumpet	16.85 10.74 9.93 8.95 13.64	14.11 7.95 8.12 6.57 10.26	15.93 9.01 9.11 7.44 12.03
Piano + Violin	14.17 9.04 8.13 10.4 9.59	12.12 7.95 6.09 9.08 7.85	11.67 7.11 5.81 8.71 7.19
Trumpet + Violin	15.40 8.87 9.14 10.51 9.17	12.49 6.23 6.87 7.92 7.77	12.13 6.31 7.17 8.11 7.95

3.5. Comparison with Other Nonnegative Factorization Models

In this section, we compare the proposed algorithm with other signal separation algorithms, in both time-domain representation and analysis the SDR results of all algorithms. The signal chosen was the same piano and trumpet mixture music used in Section 3.4. We compared the Least Square NMF (NMF-LS) and Kullback–Leibler NMF (NMF-KLD) algorithms introduced in the earlier sections of this paper, NMF with temporal continuity and sparseness criteria (NMF-TCS) [23], and NMF with automatic relevance determination (NMF-ARD) [44]. The obtained results are summarized in Table 5.

Table 5. Performance comparison of proposed method with NMF models.

Algorithm	SDR (dB)
NMF-LS	4.17
NMF-KLD	3.47
NMF-TCS	5.12
NMF-ARD	3.98
NMF using proposed method	7.63
Proposed method using matrix factor time–frequency deconvolution	12.02

When using the various NMF models, it is seen that the average improvement per source of about 3 dB has been gained by leveraging the fractional β value and adaptive sparsity. In addition, a step jump of approximately 5–8 dB in performance improvement is further obtained when the model is switched to the matrix factor time–frequency deconvolution. This is attributed to the latter model which represents both temporal structure and the pitch change which occur when an instrument plays different notes. The pitch change corresponds to a displacement on the frequency axis. Where NMF methods needed one component to model each note for each instrument, the time–frequency deconvolution model represents each instrument compactly by a single time–frequency profile convolved in both time and frequency by a time–pitch weight matrix [45]. The model dramatically decreases the number of components needed to model various instruments and effectively solves the blind single channel source separation problem for certain classes of musical signals.

4. Conclusions

This paper presents an adaptive fractional β-divergence with sparsity-aware optimization for non-negative factor time–frequency deconvolution algorithm. The impetus behind this work is that the previous β-divergence algorithms are all limited to special cases of β, and the previous sparsity methods are limited to a fixed sparsity parameter which are determined manually. Thus, these algorithms may not always produce the best results. In the proposed method, β is made adaptive and takes on fractional value. The sparsity parameter is also concurrently updated along with the estimation of β and model parameters. The convergence is theoretically proven for any β based on the auxiliary function method. This paper has shown that the proposed method is more general and can deliver better performance than other algorithms, as demonstrated using real audio recordings.

Author Contributions: W.L.W. and B.G. contributed to the mathematical development, implementation of the codes and writing of this article. A.B., B.W-K.L. and C.S.C. contributed to the simulations, technical analysis and comparison of the results.

Funding: This research was funded by National Natural Science Foundation of China (No. 61401071, No. 61527803), supported by NSAF (Grant No. U1430115) and EPSRC IAA Phase 2 funded project: "3D super-fast and portable eddy current pulsed thermography for railway inspection (EP/K503885/1).

Acknowledgments: The authors would like to thank K.Y. and S.S.D. for the helpful information on the conference edition of the work.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Mitianoudis, N.; Davies, M.E. Audio source separation: Solutions and problems. *Int. J. Adapt. Control Signal Process.* **2004**, *18*, 299–314. [CrossRef]
2. Gao, P.; Woo, W.L.; Dlay, S.S. Nonlinear signal separation for multi-nonlinearity constrained mixing model. *IEEE Trans. Neural Netw.* **2006**, *17*, 796–802. [CrossRef] [PubMed]
3. Alvarez, S.C.; Cichocki, A.; Ribas, L.C. An iterative inversion approach to blind source separation. *IEEE Trans. Neural Netw.* **2000**, *11*, 1423–1437.

4. Gao, B.; Woo, W.L.; Dlay, S.S. Single channel blind source separation using EMD-subband ariable regularized sparse features. *IEEE Trans. Audio Speech Lang. Process.* **2011**, *19*, 961–976. [CrossRef]
5. Zha, D.; Qiu, T. A new blind source separation method based on fractional lower-order statistics. *Int. J. Adapt. Control Signal Process.* **2006**, *20*, 213–223. [CrossRef]
6. Ozerov, A.; Févotte, C. Multichannel nonnegative matrix factorization in convolutive mixtures for audio source separation. *IEEE Trans. Audio Speech Lang. Process.* **2010**, *18*, 550–563. [CrossRef]
7. Zhang, J.; Woo, W.L.; Dlay, S.S. Blind source separation of post-nonlinear convolutive mixture. *IEEE Trans. Audio Speech Lang. Process.* **2007**, *15*, 2311–2330. [CrossRef]
8. Moir, T.J.; Harris, J.I. Decorrelation of multiple non-stationary sources using a multivariable crosstalk-resistant adaptive noise canceller. *Int. J. Adapt. Control Signal Process.* **2013**, *27*, 349–367. [CrossRef]
9. Djendi, M. A new two-microphone Gauss-Seidel pseudo affine projection algorithm for speech quality enhancement. *Int. J. Adapt. Control Signal Process.* **2017**, *31*, 1162–1183. [CrossRef]
10. He, X.; He, F.; Zhu, T. Large-scale super-Gaussian sources separation using Fast-ICA with rational nonlinearities. *Int. J. Adapt. Control Signal Process.* **2017**, *31*, 379–397. [CrossRef]
11. Kemiha, M.; Kacha, A. Complex blind source separation. *Circuits Syst. Signal Process.* **2017**, *36*, 1–18. [CrossRef]
12. Moazzen, I.; Agathoklis, P. A multistage space–time equalizer for blind source separation. *Circuits Syst. Signal Process.* **2016**, *35*, 185–209. [CrossRef]
13. Kumar, V.A.; Rao, C.V.R.; Dutta, A. Performance analysis of blind source separation using canonical correlation. *Circuits Syst. Signal Process.* **2018**, *37*, 658–673. [CrossRef]
14. Zhang, C.; Wang, Y.; Jing, F. Underdetermined blind source separation of synchronous orthogonal frequency hopping signals based on single source points detection. *Sensors* **2017**, *17*, 2074. [CrossRef] [PubMed]
15. Guo, Q.; Ruan, G.; Liao, Y. A time-frequency domain underdetermined blind source separation algorithm for mimo radar signals. *Symmetry* **2017**, *9*, 104. [CrossRef]
16. Li, T.; Wang, S.; Zio, E.; Shi, J.; Hong, W. Aliasing signal separation of superimposed abrasive debris based on degenerate unmixing estimation technique. *Sensors* **2018**, *18*, 866. [CrossRef] [PubMed]
17. Lee, D.; Seung, H. Learning the parts of objects by nonnegative matrix factorization. *Nature* **1999**, *401*, 788–791. [PubMed]
18. Donoho, D.; Stodden, V. *When Does Non-Negative Matrix Factorisation Give a Correct Decomposition into Parts*; MIT Press: Cambridge, MA, USA, 2004.
19. Bertin, N.; Badeau, R.; Vincent, E. Enforcing harmonicity and smoothness in Bayesian non-negative matrix factorization applied to polyphonic music transcription. *IEEE Trans. Audio Speech Lang. Process.* **2010**, *18*, 538–549. [CrossRef]
20. Vincent, E.; Bertin, N.; Badeau, R. Adaptive harmonic spectral decomposition for multiple pitch estimation. *IEEE Trans. Audio Speech Lang. Process.* **2010**, *18*, 528–537. [CrossRef]
21. Smaragdis, P. Non-negative matrix factor deconvolution; extraction of multiple sound sources from monophonic inputs. *Int. Conf. Indep. Compon. Anal. Blind Signal Sep.* **2004**, *3195*, 494–499.
22. Schmidt, M.N.; Morup, M. Nonnegative matrix factor two-dimensional deconvolution for blind single channel source separation. *Intl. Conf. Indep. Compon. Anal. Blind Signal Sep.* **2006**, *3889*, 700–707.
23. Virtanen, T. Monaural sound source separation by nonnegative matrix factorization with temporal continuity and sparseness criteria. *IEEE Trans. Audio Speech Lang. Process.* **2007**, *15*, 1066–1074. [CrossRef]
24. Laroche, C.; Papadopoulos, H.; Kowalski, M.; Richard, G. Drum extraction in single channel audio signals using multi-layer non-negative matrix factor deconvolution. In Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, New Orleans, LA, USA, 5–9 March 2017.
25. Zhi, R.C.; Flierl, M.; Ruan, Q.; Kleijn, W.B. Graph-preserving sparse nonnegative matrix factorization with application to facial expression recognition. *IEEE Trans. Syst. Man Cybern. Part B* **2011**, *41*, 38–52.
26. Okun, O.; Priisalu, H. Unsupervised data reduction. *Signal Process.* **2007**, *87*, 2260–2267. [CrossRef]
27. Kompass, R. A generalized divergence measure for nonnegative matrix factorization. *Neural Comput.* **2007**, *19*, 780–791. [CrossRef] [PubMed]
28. Cichocki, A.; Zdunek, R.; Amari, S. Csiszar's divergences for non-negative matrix factorization: Family of new algorithms. *Int. Conf. Indep. Compon. Anal. Blind Signal Sep.* **2006**, *3889*, 32–39.
29. Gao, B.; Woo, W.L.; Ling, B.W.K. Machine learning source separation using maximum a posteriori nonnegative matrix factorization. *IEEE Trans. Cybern.* **2014**, *44*, 1169–1179. [PubMed]

30. Wu, Z.; Ye, S.; Liu, J.; Sun, L.; Wei, Z. Sparse non-negative matrix factorization on GPUs for hyperspectral unmixing. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2014**, *7*, 3640–3649. [CrossRef]
31. Gao, B.; Woo, W.L.; Dlay, S.S. Adaptive sparsity non-negative matrix factorization for single-channel source separation. *IEEE J. Sel. Top. Signal Process.* **2011**, *5*, 989–1001. [CrossRef]
32. Cemgil, A.T. Bayesian inference for nonnegative matrix factorization models. *Comput. Intell. Neurosci.* **2009**. [CrossRef] [PubMed]
33. Fevotte, C.; Bertin, N.; Durrieu, J.L. Nonnegative matrix factorization with the Itakura-Saito divergence: With application to music analysis. *Neural Comput.* **2009**, *21*, 793–830. [CrossRef] [PubMed]
34. Fevotte, C.; Idier, J. Algorithms for nonnegative matrix factorization with the β-divergence. *Neural Comput.* **2010**, *23*, 2421–2456. [CrossRef]
35. Yu, K.; Woo, W.L.; Dlay, S.S. Variational regularized two-dimensional nonnegative matrix factorization with the flexible β-divergence for single channel source separation. In Proceedings of the 2nd IET International Conference in Intelligent Signal Processing (ISP), London, UK, 1–2 December 2015.
36. Gao, B.; Woo, W.L.; Dlay, S.S. Variational regularized two-dimensional nonnegative matrix factorization. *IEEE Trans. Neural Netw. Learn. Syst.* **2012**, *23*, 703–716. [PubMed]
37. Parathai, P.; Woo, W.L.; Dlay, S.S. Single-channel blind separation using L1-sparse complex nonnegative matrix factorization for acoustic signals. *J. Acoust. Soc. Am.* **2015**. [CrossRef] [PubMed]
38. Tengtrairat, N.; Woo, W.L.; Dlay, S.S.; Gao, B. Online noisy single-channel blind separation by spectrum amplitude estimator and masking. *IEEE Trans. Signal Process.* **2016**, *64*, 1881–1895. [CrossRef]
39. Tengtrairat, N.; Gao, B.; Woo, W.L.; Dlay, S.S. Single-channel blind separation using pseudo-stereo mixture and complex 2-D histogram. *IEEE Trans. Neural Netw. Learn. Syst.* **2013**, *24*, 1722–1735. [CrossRef] [PubMed]
40. Goto, M.; Hashiguchi, H.; Nishimura, T.; Oka, R. RWC music database: Music genre database and musical instrument sound database. In Proceedings of the International Symposium on Music Information Retrieval, Baltimore, MD, USA, 26–30 October 2003.
41. Vincent, A.; Gribonval, R.; Fevotte, C. Performance measurement in blind audio source separation. *IEEE Trans. Speech Audio Process.* **2005**, *14*, 1462–1469. [CrossRef]
42. Signal Separation Evaluation Campaign (SiSEC 2018). 2018. Available online: http://sisec.wiki.irisa.fr (accessed on 22 April 2018).
43. Rabiner, L.R. A tutorial on hidden Markov models and selected applications in speech recognition. *Proc. IEEE* **1989**, *77*, 257–286. [CrossRef]
44. Mørup, M.; Hansen, K.L. Tuning pruning in sparse non-negative matrix factorization. In Proceedings of the 17th European Signal Processing Conference (EUSIPCO'09), Glasgow, Scotland, 24–28 August 2009.
45. Al-Tmeme, A.; Woo, W.L.; Dlay, S.S.; Gao, B. Underdetermined convolutive source separation using GEM-MU with variational approximated optimum model order NMF2D. *IEEE Trans. Audio Speech Lang. Process.* **2017**, *25*, 35–49. [CrossRef]