

METHOD AND AUDIO PROCESSING SYSTEM FOR BLIND SOURCE SEPARATION WITHOUT SAMPLING RATE MISMATCH ESTIMATION AND COMPENSATION

BACKGROUND

Field of Invention

[0001] The present disclosure relates to asynchronous recordings; in particular, there is no need to estimate sampling rate mismatch and resample the audio signals.

Description of Related Art

[0002] The use of an ad-hoc microphone array is becoming increasingly useful in the field of audio signal processing. For blind source separation, an ad-hoc microphone array offers an affordable, flexible, and portable audio acquisition system. However, the signals acquired by ad-hoc microphone array often have different starting times. Furthermore, since the analog-to-digital converters (ADC) between the recording devices are independent of one another, the true sampling rate on each device varies slightly off the nominal sampling rate. These variations can cause significant degradation in the performance of conventional signal processing algorithms, such as blind source separation, thus require correction. While pre-recording calibration approaches exist, these are not always possible or convenient on most recording devices. Hence, in practice, the compensation of the sampling rate mismatch and starting time offset is often done post-recording where no prior information about the setup or the mismatch is known.

SUMMARY

[0003] Embodiments of the present disclosure provide a method for blind source separation on an audio processing system with multiple devices, in which each of the devices includes multiple microphones. The method includes: (a) obtaining an audio signal sensed by each of the microphones of each of the devices, dividing each of the audio signals into multiple time frames, and performing time to frequency transformation on each of the time frames to generate a vector $\mathbf{x}_d[\tau, k; f_d]$ in a time-frequency (TF) domain, where d denotes the device index, f_d denotes a sampling rate of the d^{th} device, τ denotes the time frame index, and k the frequency bin index; (b) setting a mixing matrix $\mathbf{A}[k] = [\mathbf{A}_1[k; f_1]^T, \dots, \mathbf{A}_D[k; f_D]^T]^T$, where D denotes the number of the devices, the matrix $\mathbf{A}_d[k; f_d]$ is the mixing matrix of the d^{th} device such that the size of the mixing matrix $\mathbf{A}_d[k; f_d]$ is $c_d \times N$, with c_d being the number of the microphones of the d^{th} device, and N is the number of multiple signal sources; (c) for each of the signal sources, each of the devices, and each of the time frames, computing a measure of dissimilarity between the vector $\mathbf{x}_d[\tau, k; f_d]$ and the vector $\mathbf{a}_{n,d}[k; f_d]$ which denotes the n^{th} column of the matrix $\mathbf{A}_d[k; f_d]$, in which n is a positive integer less than or equal to N ; (d) establishing an objective function according to the measures of dissimilarity, and performing an optimization algorithm according to the objective function to compute the mixing matrix; and (e) computing multiple estimates of the original signals from the input signals according to the mixing matrix and the vectors $\mathbf{x}_d[\tau, k; f_d]$ of the devices without estimating a sampling rate mismatch between the devices.

[0004] In some embodiments, the measure of dissimilarity is the cosine distance between the vector $\mathbf{x}_d[\tau, k; f_d]$ and the vector $\mathbf{a}_{n,d}[k; f_d]$.

[0005] In some embodiments, the method further includes: computing a discriminative term based on the measure of dissimilarity, in which the discriminative term, $z_d[\tau, k]$, is represented as the following equation where r is a real number, and $\mathbf{a}_{n,d}^H[k; f_d]$ is a conjugate transpose of the vector $\mathbf{a}_{n,d}[k; f_d]$:

$$z_d[\tau, k] = \left[\frac{1}{N} \sum_{n=1}^N \left[\left(1 - \frac{|\mathbf{a}_{n,d}^H[k; f_d] \mathbf{x}_d[\tau, k; f_d]|^2}{\|\mathbf{a}_{n,d}[k; f_d]\|_2^2 \|\mathbf{x}_d[\tau, k; f_d]\|_2^2} \right) \right]^r \right]^{1/r}$$

[0006] In some embodiments, the objective function is represented as the following equation in which T is the number of the time frames.

$$\min_{\mathbf{A}_1[k; f_1], \dots, \mathbf{A}_D[k; f_D]} J^{(\text{AsyncDSF-MIN})} = \sum_{\tau=1}^T \min(z_1[\tau, k], \dots, z_d[\tau, k], \dots, z_D[\tau, k])$$

[0007] In some embodiments, the step of performing the optimization algorithm to update the mixing matrix includes: (i) computing a sub-gradient of the objective function using analytic gradient or automatic differentiation; and (ii1) subtracting a product of the sub-gradient and a learning rate from the mixing matrix plus an optional momentum term or (ii2) subtracting a product of the sub-gradient and an approximation of the Hessian matrix from the mixing matrix.

$$\frac{\partial J^{(\text{AsyncDSF-MIN})}}{\partial \mathbf{a}_{n,d}^*[k; f_d]} = \sum_{\tau=1}^T \frac{\partial \min(z_1[\tau, k], \dots, z_D[\tau, k])}{\partial z_d[\tau, k]} \frac{\partial z_d[\tau, k]}{\partial \mathbf{a}_{n,d}^*[k; f_d]}$$

$$\frac{\partial \min(z_1[\tau, k], \dots, z_D[\tau, k])}{\partial z_d[\tau, k]} = \begin{cases} 1, & z_d[\tau, k] = \min(z_1[\tau, k], \dots, z_D[\tau, k]) \\ 0, & \text{otherwise} \end{cases}$$

$$\frac{\partial z_d[\tau, k]}{\partial \mathbf{a}_{n,d}^*[k; f_d]} = \left(\frac{\mathbf{g}_{n,d}[\tau, k]}{\|\mathbf{a}_{n,d}[k; f_d]\|_2} - \frac{\mathbf{a}_{n,d}[k; f_d] \Re\{\mathbf{a}_{n,d}^H[k; f_d] \mathbf{g}_{n,d}[\tau, k]\}}{\|\mathbf{a}_{n,d}[k; f_d]\|_2^3} \right)$$

$$\mathbf{g}_{n,d}[\tau, k] = 2 \frac{\mathbf{x}_d[\tau, k; f_d] (\mathbf{x}_d^H[\tau, k; f_d] \mathbf{a}_{n,d}[k; f_d]) h_{n,d}[\tau, k]}{\|\mathbf{x}_d[\tau, k; f_d]\|_2^2 \|\mathbf{a}_{n,d}[k; f_d]\|_2}$$

$$h_{n,d}[\tau, k] = -\frac{1}{N} \left(\frac{z_d[\tau, k]}{1 - \frac{|\mathbf{a}_{n,d}^H[k; f_d] \mathbf{x}_d[\tau, k; f_d]|^2}{\|\mathbf{a}_{n,d}[k; f_d]\|_2^2 \|\mathbf{x}_d[\tau, k; f_d]\|_2^2}} \right)^{1-r}$$

[0008] In some embodiments, the objective function is represented as the following equation.

$$\min_{\mathbf{A}_1[k; f_1], \dots, \mathbf{A}_D[k; f_D]} J^{(\text{AsyncDSF-MAX})} = \sum_{\tau=1}^T \max_d z_d[\tau, k]$$

[0009] In some embodiments, the step of performing the optimization algorithm to update the mixing matrix includes: (i) computing a sub-gradient of the objective function using analytic gradient or automatic differentiation; and (ii1) subtracting a product of the sub-gradient and a learning rate from the mixing matrix plus an optional momentum term or (ii2) subtracting a product of the sub-gradient and an approximation of the Hessian matrix from the mixing matrix.

$$\frac{\partial J^{(\text{AsyncDSF-MAX})}}{\partial \mathbf{a}_{n,d}^*[k; f_d]} = \sum_{\tau=1}^T \frac{\partial \max(z_1[\tau, k], \dots, z_D[\tau, k])}{\partial z_d[\tau, k]} \frac{\partial z_d[\tau, k]}{\partial \mathbf{a}_{n,d}^*[k; f_d]}$$

where

$$\frac{\partial \max(z_1[\tau, k], \dots, z_D[\tau, k])}{\partial z_d[\tau, k]} = \begin{cases} 1, & z_d[\tau, k] = \max(z_1[\tau, k], \dots, z_D[\tau, k]) \\ 0, & \text{otherwise} \end{cases}$$

$$\frac{\partial z_d[\tau, k]}{\partial \mathbf{a}_{n,d}^*[k; f_d]} = \left(\frac{\mathbf{g}_{n,d}[\tau, k]}{\|\mathbf{a}_{n,d}[k; f_d]\|_2} - \frac{\mathbf{a}_{n,d}[k; f_d] \Re\{\mathbf{a}_{n,d}^H[k; f_d] \mathbf{g}_{n,d}[\tau, k]\}}{\|\mathbf{a}_{n,d}[k; f_d]\|_2^3} \right)$$

$$\mathbf{g}_{n,d}[\tau, k] = 2 \frac{\mathbf{x}_d[\tau, k; f_d] (\mathbf{x}_d^H[\tau, k; f_d] \mathbf{a}_{n,d}[k; f_d]) h_{n,d}[\tau, k]}{\|\mathbf{x}_d[\tau, k; f_d]\|_2^2 \|\mathbf{a}_{n,d}[k; f_d]\|_2}$$

$$h_{n,d}[\tau, k] = -\frac{1}{N} \left(\frac{z_d[\tau, k]}{1 - \frac{|\mathbf{a}_{n,d}^H[k; f_d] \mathbf{x}_d[\tau, k; f_d]|^2}{\|\mathbf{a}_{n,d}[k; f_d]\|_2^2 \|\mathbf{x}_d[\tau, k; f_d]\|_2^2}} \right)^{1-r}$$

[0010] In some embodiments, the step of computing the original signals of the signal sources according to the mixing matrix and the vectors $\mathbf{x}_d[\tau, k; f_d]$ of the device includes: constituting a vector $\mathbf{x}[\tau, k] = [\mathbf{x}_1[\tau, k; f_1]^T, \dots, \mathbf{x}_D[\tau, k; f_D]^T]^T \in \mathbb{C}^C$ where $C = \sum c_d$ and D denotes the number of the devices; and multiplying an inverse of mixing matrix by the vector $\mathbf{x}[\tau, k]$ to obtain the original signals or multiplying a time-frequency mask derived from the mixing matrix and $\mathbf{x}[\tau, k]$ by $\mathbf{x}[\tau, k]$ to obtain the original signals.

[0011] From another aspect, embodiments of the present disclosure provide an audio processing system including multiple devices and a server. Each of the devices includes multiple microphones. Each of the microphones of each of the devices is configured to sense an audio signal. The devices and the server are configured to perform steps of: (a) obtaining an audio signal sensed by each of the microphones of each of the devices, dividing each of the audio signals into multiple time frames, and performing time to frequency transformation on each of the time frames to generate a vector $\mathbf{x}_d[\tau, k; f_d]$ in a time-frequency (TF) domain, where d denotes the device index, f_d denotes a sampling rate of the

d^{th} device, τ denotes time frame index, and k the frequency bin index; (b) setting a mixing matrix $\mathbf{A}[\mathbf{k}] = [\mathbf{A}_1[k; f_1]^T, \dots, \mathbf{A}_D[k; f_D]^T]^T$, in which D denotes the number of the devices, the matrix $\mathbf{A}_d[k; f_d]$ is the mixing matrix of the d^{th} device such that the size of the mixing matrix $\mathbf{A}_d$ is $c_d \times N$, with c_d being the number of the microphones of the d^{th} device, and N is the number of multiple signal sources; (c) for each of the signal sources, each of the devices, and each of the time frames, computing a measure of dissimilarity between the vector $\mathbf{x}_d[\tau, k; f_d]$ and the vector $\mathbf{a}_{n,d}[k; f_d]$ which denotes the n^{th} column of the matrix $\mathbf{A}_d[k; f_d]$, in which n is a positive integer less than or equal to N ; (d) establishing an objective function according to the measure of dissimilarity, and performing an optimization algorithm according to the objective function to compute the mixing matrix; and (e) computing multiple original signals of the signal sources according to the mixing matrix and the vectors $\mathbf{x}_d[\tau, k; f_d]$ of the devices without estimating a sampling rate mismatch between the devices.

[0012] In some embodiments, the devices are configured to transmit the audio signals to the server which is configured to perform the steps (a) to (e).

[0013] In some embodiments, the devices are configured to perform the steps (a) to (c) to calculate multiple discriminative terms and transmit the discriminative terms to the server. The server is configured to compute at least one frame index according to the discriminative terms and transmit at least one frame index to the devices to update the mixing matrix.

BRIEF DESCRIPTION OF THE DRAWINGS

[0014] The invention can be more fully understood by reading the following detailed description of the embodiment, with reference made to the accompanying drawings as follows.

[0015] FIG. 1 is a schematic diagram of an audio processing system in accordance with some embodiments.

[0016] FIG. 2 is a flow chart of the centralized scheme in accordance with some embodiments.

[0017] FIG. 3 is a flow chart of the distributed scheme in accordance with some embodiments.

[0018] FIG. 4 is a schematic diagram of data transmission between the server and the devices in the distributed scheme in accordance with some embodiments.

DETAILED DESCRIPTION

[0019] Specific embodiments of the present invention are further described in detail below with reference to the accompanying drawings, however, the embodiments described are not intended to limit the present invention and it is not intended for the description of the operation to limit the order of implementation. Moreover, any device with equivalent functions that are produced from a structure formed by recombination of elements shall fall within the scope of the present invention. Additionally, the drawings are only illustrative and are not drawn to actual size.

[0020] The use of “first”, “second”, “third”, etc. in the specification should be understood for identifying units or data described by the same terminology but are not referred to in any particular order or sequence.

[0021] FIG. 1 is a schematic diagram of an audio processing system in accordance with some embodiments. Referring to FIG. 1, an audio processing system 100 includes devices 101-103 and a server 120. The devices 101-103 may be smartphones, tablets, laptops, or any electrical device with recording functions. Each of the devices 101-103 includes more than one microphone, and each microphone represents a channel. Each of users 111-113 represents a signal source. Herein, there are D devices in which a signal sensed by the d^{th} device is represented as a vector $\mathbf{x}_d = [x_{d,1}[n], \dots, x_{d,c_d}[n]]^T$. $x_{d,c}[n]$ denotes the n^{th} sample in discrete time at the c^{th} channel of the d^{th} device. D , d , c , and n are positive integers. The sampling rate of the first device is f_1 , and the continuous-to-discrete sampling of the first device is represented as the following equation (1). The sampling rate mismatch between the first device and the d^{th} device occurs as described in equation (2).

$$\mathbf{x}_1[n] = \mathbf{x}_1\left(\frac{n}{f_1}\right) \quad (1)$$

$$\mathbf{x}_d[n] = \mathbf{x}_d\left(\frac{n+\delta_d}{(1+\epsilon_d)f_1}\right) \quad (2)$$

[0022] $\mathbf{x}_1(\cdot)$ denotes the continuous-time signal of the first device, and $\mathbf{x}_1[\cdot]$ denotes the discrete-time signal of the first device. ϵ_d is the sampling rate mismatch of the d^{th} device with respect to the first device and δ_d is the discrete-time offset between the start of the recording of the d^{th} device and the

first device. The start-time offset δ_d can be computed by any algorithms such as the following equation (3).

$$\delta_d \approx \operatorname{argmax}_{\delta_d} \sum_j (x_{1,j} \star x_{d,j})[\delta_d] \quad (3)$$

[0023] “ $\star$ ” is the cross-correlation operator. After the start-time offset δ_d is estimated, the audio signal can be compensated by the following equation (4).

$$x_d[n] \leftarrow x_d[n - \delta_d] \quad (4)$$

[0024] In the conventional algorithms, the sampling rate mismatch ϵ_d is estimated, and then the audio signals are resampled, but this approach is computationally expensive. In the disclosure, the sampling rate mismatch ϵ_d estimate and resampling are not required. We assume that each of the devices 101-103 includes at least two channels and that the channels within the same device are sampled at the same rate. Under this assumption, we can infer, from each device separately, spatially discriminative terms which are independent of the sampling rate mismatch. Using the discriminative terms, we are then able to aggregate the information across the devices and learn the mixing matrix of all devices. Once the mixing matrix is learned, estimates of the original signals of the sources can be computed.

[0025] Consider an ad-hoc microphone array comprising a set of D devices, where the d^{th} device includes $c_d \geq 2$ channels (i.e., microphones). We assume that, for each device, there is no sampling rate mismatch between the channels. Hence, the sampling rate mismatch only occurs between the devices but not between the channels of the same device. We denote $C = \sum c_d$ as the

total number of the channels. Due to reverberation, the recorded audio signals at the devices are convolutive.

[0026] First, an audio signal sensed by each microphone of each device is obtained. Each audio signal is divided into several time frames and the length thereof is not limited. A time-frequency transform is performed to each time frame to generate a vector $\mathbf{x}_d[\tau, k; f_d]$ in a time-frequency (TF) domain. The said time-frequency transform is, for example, the short time Fourier transform (STFT). The vector $\mathbf{x}_d[\tau, k; f_d]$ includes audio signals sensed by c_d channels of the d^{th} device where $\mathbf{x}_d[\tau, k; f_d] = [x_{d,1}[\tau, k; f_d], \dots, x_{d,c_d}[\tau, k; f_d]]^T \in \mathbb{C}^{c_d}$. $x_{d,c}[\tau, k; f_d]$ is the audio signal, after STFT is performed, at the k^{th} frequency bin index of the τ^{th} time frame on the c^{th} channel of the d^{th} device where f_d is the sampling rate of the d^{th} device. We consider the case where the τ^{th} time frame corresponds to roughly the same continuous-time region under the assumption that the recording length is short or the sampling rate mismatch is small, such that the original signals in that time frame across all devices are approximately the same. The vector $\mathbf{x}_d[\tau, k; f_d]$ is generated by the product of the mixing matrix and the original signals in TF domain as shown in the following equation (5).

$$\mathbf{x}_d[\tau, k; f_d] = \mathbf{A}_d[k; f_d] \mathbf{s}_d[\tau, k; f_d] \quad (5)$$

[0027] The size of the mixing matrix $\mathbf{A}_d[k; f_d]$ is $c_d \times N$ where N is the number of all signal sources 111-113. The vector $\mathbf{s}_d[\tau, k; f_d] \in \mathbb{C}^N$ includes the original signals of the same N signal sources that are sampled by the d^{th} device in TF domain at different sampling rates. We can assemble multiple vectors/matrices across all devices as per the following equations (6) to (8).

$$\mathbf{x}[\tau, k] = [\mathbf{x}_1[\tau, k; f_1]^T, \dots, \mathbf{x}_D[\tau, k; f_D]^T]^T \in \mathbb{C}^C \quad (6)$$

$$\mathbf{A}[k] = [\mathbf{A}_1[k; f_1], \dots, \mathbf{A}_D[k; f_D]] \in \mathbb{C}^{C \times ND} \quad (7)$$

$$\mathbf{s}[\tau, k] = [\mathbf{s}_1[\tau, k; f_1]^T, \dots, \mathbf{s}_D[\tau, k; f_D]^T]^T \in \mathbb{C}^{ND} \quad (8)$$

[0028] Therefore, the equation (5) is rewritten as the following equation (9).

$$\mathbf{x}[\tau, k] = \mathbf{A}[k]\mathbf{s}[\tau, k] \quad (9)$$

[0029] The length of the vector $\mathbf{x}[\tau, k]$ is C , the size of the matrix $\mathbf{A}[k]$ is $C \times ND$, and the length of the vector $\mathbf{s}[\tau, k]$ is ND . In the embodiment, the mixing matrix $\mathbf{A}[k]$ is to be computed, and then the original signals $\mathbf{s}[\tau, k]$ are computed according to the mixing matrix $\mathbf{A}[k]$ and the vector $\mathbf{x}[\tau, k]$, for example, by multiplying the inverse of the mixing matrix $\mathbf{A}[k]$ by the vector $\mathbf{x}[\tau, k]$ or by multiplying a time-frequency mask derived from the mixing matrix and $\mathbf{x}[\tau, k]$ by the vector $\mathbf{x}[\tau, k]$. Accordingly, there is no need to estimate the sampling rate mismatch.

[0030] In the embodiment, for each signal source n , each device d , and each time frame k , a measure of dissimilarity between the vector $\mathbf{x}_d[\tau, k; f_d]$ and a vector $\mathbf{a}_{n,d}[k; f_d]$ is computed. The measure of dissimilarity is, for example, cosine distance but may be Euclidean distance or other suitable measure in other embodiments. The vector $\mathbf{a}_{n,d}[k; f_d]$ is the n^{th} column of the matrix $\mathbf{A}_d[k; f_1]$ where n is a positive integer less than or equal to N . A discriminative term is computed according to the measure of dissimilarity as the following equation (10).

$$z_d[\tau, k] = \left[\frac{1}{N} \sum_{n=1}^N \left[\left(1 - \frac{|\mathbf{a}_{n,d}^H[k; f_d] \mathbf{x}_d[\tau, k; f_d]|^2}{\|\mathbf{a}_{n,d}[k; f_d]\|_2^2 \|\mathbf{x}_d[\tau, k; f_d]\|_2^2} \right) \right]^r \right]^{1/r} \quad (10)$$

[0031] r is a real number which can be determined by experiments. $\mathbf{a}_{n,d}^H[k; f_d]$ is the conjugate transpose of the vector $\mathbf{a}_{n,d}[k; f_d]$. An objective function is established according to the discriminative term. An optimization algorithm is performed according to the objective function to compute the mixing matrix $\mathbf{A}[k]$. Two embodiments are provided below in detail.

[0032] In the first embodiment, the minimum discriminative term from all D devices is used to establish an objective function $J^{(\text{AsyncDSF-MIN})}$ as per the following equation (11).

$$\min_{\mathbf{A}_1[k; f_1], \dots, \mathbf{A}_D[k; f_D]} J^{(\text{AsyncDSF-MIN})} = \sum_{\tau=1}^T \min_d z_d[\tau, k] \quad (11)$$

[0033] T is the number of the time frames. In the other embodiment, the maximum discriminative term from all D devices is used to establish an objective function $J^{(\text{AsyncDSF-MAX})}$ as the following equation (12).

$$\min_{\mathbf{A}_1[k; f_1], \dots, \mathbf{A}_D[k; f_D]} J^{(\text{AsyncDSF-MAX})} = \sum_{\tau=1}^T \max_d z_d[\tau, k] \quad (12)$$

[0034] In other embodiments, an average of all discriminative terms from the D devices is used to establish an objective function which is not written for simplicity. Note that the objective functions $J^{(\text{AsyncDSF-MIN})}$ and $J^{(\text{AsyncDSF-MAX})}$ require only local information at each device which is synchronized by default (since ADCs on the same device use a common clock signal, this is not true across different devices). Physically, the discriminative term $z_d[\tau, k]$ becomes zero when the mixing matrix $\mathbf{A}[k]$ is correctly estimated. On the other hand, the discriminative term $z_d[\tau, k]$ will be of a large value if the mixing matrix $\mathbf{A}[k]$ is not correct. Intuitively, we want the discriminative term $z_d[\tau, k]$ to be small for all time frames and frequency bin indices. When solving the equation (11) and (12),

the sub-gradient of the objective function is computed as the following equations (13) and (14) respectively.

$$\frac{\partial J^{(\text{AsyncDSF-MIN})}}{\partial \mathbf{a}_{n,d}^*[k;f_d]} = \sum_{\tau=1}^T \frac{\partial \min(z_1[\tau,k], \dots, z_D[\tau,k])}{\partial z_d[\tau,k]} \frac{\partial z_d[\tau,k]}{\partial \mathbf{a}_{n,d}^*[k;f_d]} \quad (13)$$

$$\frac{\partial J^{(\text{AsyncDSF-MAX})}}{\partial \mathbf{a}_{n,d}^*[k;f_d]} = \sum_{\tau=1}^T \frac{\partial \max(z_1[\tau,k], \dots, z_D[\tau,k])}{\partial z_d[\tau,k]} \frac{\partial z_d[\tau,k]}{\partial \mathbf{a}_{n,d}^*[k;f_d]} \quad (14)$$

where

$$\frac{\partial \max(z_1[\tau, k], \dots, z_D[\tau, k])}{\partial z_d[\tau, k]} = \begin{cases} 1, & z_d[\tau, k] = \max(z_1[\tau, k], \dots, z_D[\tau, k]) \\ 0, & \text{otherwise} \end{cases}$$

$$\frac{\partial \min(z_1[\tau, k], \dots, z_D[\tau, k])}{\partial z_d[\tau, k]} = \begin{cases} 1, & z_d[\tau, k] = \min(z_1[\tau, k], \dots, z_D[\tau, k]) \\ 0, & \text{otherwise} \end{cases}$$

$$\frac{\partial z_d[\tau, k]}{\partial \mathbf{a}_{n,d}^*[k;f_d]} = \left(\frac{\mathbf{g}_{n,d}[\tau, k]}{\|\mathbf{a}_{n,d}[k;f_d]\|_2} - \frac{\mathbf{a}_{n,d}[k;f_d] \Re\{\mathbf{a}_{n,d}^H[k;f_d] \mathbf{g}_{n,d}[\tau, k]\}}{\|\mathbf{a}_{n,d}[k;f_d]\|_2^3} \right)$$

$$\mathbf{g}_{n,d}[\tau, k] = 2 \frac{\mathbf{x}_d[\tau, k; f_d] (\mathbf{x}_d^H[\tau, k; f_d] \mathbf{a}_{n,d}[k; f_d]) h_{n,d}[\tau, k]}{\|\mathbf{x}_d[\tau, k; f_d]\|_2^2 \|\mathbf{a}_{n,d}[k; f_d]\|_2}$$

$$h_{n,d}[\tau, k] = -\frac{1}{N} \left(\frac{z_d[\tau, k]}{1 - \frac{|\mathbf{a}_{n,d}^H[k; f_d] \mathbf{x}_d[\tau, k; f_d]|^2}{\|\mathbf{a}_{n,d}[k; f_d]\|_2^2 \|\mathbf{x}_d[\tau, k; f_d]\|_2^2}} \right)^{1-r}$$

[0035] The mixing matrix $\mathbf{A}[k]$ is updated for several iterations. In each iteration, a product of the sub-gradient and the learning rate is subtracted from the mixing matrix plus an optional momentum term where the learning rate can be determined by experiments. Or in each iteration, a product of the sub-gradient and an approximation of the Hessian matrix is subtracted from the mixing matrix. Note that the sub-gradient of min and max function are either 0 or 1. This

implies that in each iteration, only a subset of time frames where the above sub-gradient is non-zero is used to update the matrix $A_d[k; f_d]$. When adopting the equations (11) and (13), the matrixes $A_d[k; f_d]$ are updated using time frames where the discriminative terms are the smallest among the devices; when adopting the equations (12) and (14), the matrixes $A_d[k; f_d]$ are updated using time frames where the discriminative terms are the highest among the devices. In some experiments, the result of equations (12) and (14) is better than the other one.

[0036] In brief, the proposed approaches compute discriminative terms locally on each device. Therefore, the compensation of the sampling rate mismatch is not required. Furthermore, the update of the mixing matrix only utilizes a subset of the time frames, and therefore the proposed methods require less computational power and transmission bandwidth. This allows for real-time processing.

[0037] Two schemes, a centralized scheme, and a distributed scheme are provided in the following embodiments. In the centralized scheme, the devices 101-103 transmit the audio signals to server 120 which performs the blind source separation. For example, FIG. 2 is a flow chart of the centralized scheme in accordance with some embodiments. Referring to FIG. 2, the audio signals are gathered from the devices 101-103. Steps 201-215 are performed by server 120. In Step 201, the start-time offset is estimated and compensated. The STFT is performed in Step 202, and the transformed data is stored in a buffer 230. In Step 203, the mixing matrix is updated. Step 203 includes Steps 204-213.

[0038] In Step 204, a whitening transform such as principle component analysis (PCA) is performed on the audio signals. In Step 205, normalization such as L2-

norm normalization is performed. In Step 206, the mixing matrix is initialized. Step 207 includes Steps 208-211. In Step 208, flattening is performed to transform the mixing matrix into a long vector. In Step 209, a cost function (i.e. the objective function) is established. In Step 210, a gradient-based optimizer is used to update the vector. In Step 211, unflatten is performed to transform the updated vector into a matrix. In Step 212, a soft mask is computed. To be specific, the computation of Step 212 is written as the following equations (15) and (16).

$$\mathbf{y}_{d,n}[\tau, k] = \mathbf{M}_d[\tau, k, n] \circ \mathbf{x}_d[\tau, k] \quad (15)$$

$$\mathbf{M}_d[\tau, k, n] = \frac{\exp[\beta(1-\mathbf{x}_{d,n}[\tau, k])]}{\sum_{n=1}^N \exp[\beta(1-\mathbf{x}_{d,n}[\tau, k])]} \quad (16)$$

[0039] The operator “ $\circ$ ” represents the Hadamard elementwise multiplication. β is a real number such as 12.5. In Step 213, permutation alignment of source indices between frequency bins is performed. In Step 214, the source separation is performed. That is, the original signals are computed according to the audio signals and the mixing matrix. In Step 215, inverse STFT is performed to obtain the original signals in the time domain.

[0040] FIG. 3 is a flow chart of the distributed scheme in accordance with some embodiments. Referring to the FIG. 3, Steps 201-206, 301, and 212-215 are performed by the devices 101-103. The difference between FIG. 2 and FIG. 3 is Step 301 which includes steps 302 and 303. Referring to FIG. 3 and FIG. 4, in the 302, the devices 101-103 compute the discriminative terms locally, and transmit the discriminative terms to the server 120. The server 120 establishes the cost function according to the discriminative terms, performs the optimization algorithm to obtain frame indices of the time frames to be updated. In Step 303,

the devices 101-103 receive the frame indices from the server 120, and update the mixing matrix according to the frame indices. Note that the data bandwidth required in the centralized scheme is proportional to the number of the channels, but the bandwidth is reduced in the distributed scheme. For example, if each device has 6 microphones (i.e., channels), then the data bandwidth is reduced by 6 times in the distributed scheme.

[0041] The proposed blind source separation may be applied to any technical field such as audio enhancement, voice identification, text generation, natural language processing, and speech analysis which is not limited in the disclosure.

[0042] Although the present invention has been described in considerable detail with reference to certain embodiments thereof, other embodiments are possible. Therefore, the spirit and scope of the appended claims should not be limited to the description of the embodiments contained herein. It will be apparent to those skilled in the art that various modifications and variations can be made to the structure of the present invention without departing from the scope or spirit of the invention. In view of the foregoing, it is intended that the present invention cover modifications and variations of this invention provided they fall within the scope of the following claims.

WHAT IS CLAIMED IS:

1. A method for blind source separation on an audio processing system comprising a plurality of devices, wherein each of the devices comprises a plurality of microphones, and the method comprises:

obtaining an audio signal sensed by each of the microphones of each of the devices, dividing each of the audio signals into a plurality of time frames, and performing time to frequency transform on each of the time frames to generate a vector $\mathbf{x}_d[\tau, k; f_d]$ in a time-frequency (TF) domain, wherein d denotes a d^{th} device, f_d denotes a sampling rate of the d^{th} device, τ denotes a τ^{th} time frame, and k denotes a k^{th} frequency bin index;

setting a mixing matrix $\mathbf{A}[\mathbf{k}] = [\mathbf{A}_1[k; f_1]^T, \dots, \mathbf{A}_D[k; f_D]^T]^T$, wherein D denotes a number of the devices, a matrix $\mathbf{A}_d[k; f_d]$ corresponds to the d^{th} device, the size of the mixing matrix $\mathbf{A}_d$ is $c_d \times N$, with c_d being a number of the microphones of the d^{th} device, and N is a number of a plurality of signal sources;

for each of the signal sources, each of the devices, and each of the time frames, computing a measure of dissimilarity between the vector $\mathbf{x}_d[\tau, k; f_d]$ and a vector $\mathbf{a}_{n,d}[k; f_d]$ which denotes a n^{th} column of the matrix $\mathbf{A}_d[k; f_d]$, wherein n is a positive integer less than or equal to N ;

establishing an objective function according to the measures of dissimilarity, and performing an optimization algorithm according to the objective function to compute the mixing matrix; and

computing a plurality of original signals of the signal sources according to the mixing matrix and the vectors $\mathbf{x}_d[\tau, k; f_d]$ of the devices without estimating a sampling rate mismatch between the devices.

2. The method of claim 1, wherein the measure of dissimilarity is a cosine distance between the vector $\mathbf{x}_d[\tau, k; f_d]$ and the vector $\mathbf{a}_{n,d}[k; f_d]$.

3. The method of claim 2, further comprising:

computing a discriminative term based on the measure of dissimilarity, wherein the discriminative term is represented as a following equation,

$$z_d[\tau, k] = \left[\frac{1}{N} \sum_{n=1}^N \left[\left(1 - \frac{|\mathbf{a}_{n,d}^H[k; f_d] \mathbf{x}_d[\tau, k; f_d]|^2}{\|\mathbf{a}_{n,d}[k; f_d]\|_2^2 \|\mathbf{x}_d[\tau, k; f_d]\|_2^2} \right) \right]^r \right]^{1/r}$$

wherein r is a real number, and $\mathbf{a}_{n,d}^H[k; f_d]$ is a conjugate transpose of the vector $\mathbf{a}_{n,d}[k; f_d]$.

4. The method of claim 3, wherein the objective function is represented as a following equation,

$$\min_{\mathbf{A}_1[k; f_1], \dots, \mathbf{A}_D[k; f_D]} J^{(\text{AsyncDSF-MIN})} = \sum_{\tau=1}^T \min_d z_d[\tau, k]$$

wherein T is a number of the time frames.

5. The method of claim 4, wherein performing the optimization algorithm comprises:

computing a sub-gradient of the objective function as per following equation,

$$\frac{\partial J^{(\text{AsyncDSF-MIN})}}{\partial \mathbf{a}_{n,d}^*[k; f_d]} = \sum_{\tau=1}^T \frac{\partial \min(z_1[\tau, k], \dots, z_D[\tau, k])}{\partial z_d[\tau, k]} \frac{\partial z_d[\tau, k]}{\partial \mathbf{a}_{n,d}^*[k; f_d]}$$

where

$$\frac{\partial \min(z_1[\tau, k], \dots, z_D[\tau, k])}{\partial z_d[\tau, k]} = \begin{cases} 1, & z_d[\tau, k] = \min(z_1[\tau, k], \dots, z_D[\tau, k]) \\ 0, & \text{otherwise} \end{cases}$$

$$\frac{\partial z_d[\tau, k]}{\partial \mathbf{a}_{n,d}^*[k; f_d]} = \left(\frac{\mathbf{g}_{n,d}[\tau, k]}{\|\mathbf{a}_{n,d}[k; f_d]\|_2} - \frac{\mathbf{a}_{n,d}[k; f_d] \Re\{\mathbf{a}_{n,d}^H[k; f_d] \mathbf{g}_{n,d}[\tau, k]\}}{\|\mathbf{a}_{n,d}[k; f_d]\|_2^3} \right)$$

$$\mathbf{g}_{n,d}[\tau, k] = 2 \frac{\mathbf{x}_d[\tau, k; f_d] (\mathbf{x}_d^H[\tau, k; f_d] \mathbf{a}_{n,d}[k; f_d]) h_{n,d}[\tau, k]}{\|\mathbf{x}_d[\tau, k; f_d]\|_2^2 \|\mathbf{a}_{n,d}[k; f_d]\|_2}$$

$$h_{n,d}[\tau, k] = -\frac{1}{N} \left(\frac{z_d[\tau, k]}{1 - \frac{|\mathbf{a}_{n,d}^H[k; f_d] \mathbf{x}_d[\tau, k; f_d]|^2}{\|\mathbf{a}_{n,d}[k; f_d]\|_2^2 \|\mathbf{x}_d[\tau, k; f_d]\|_2^2}} \right)^{1-r}$$

; and

subtracting a product of the sub-gradient and a learning rate to the mixing matrix plus an optional momentum term to update the mixing matrix

, or

subtracting a product of the sub-gradient and an approximation of the Hessian matrices from the mixing matrix.

6. The method of claim 3, wherein the objective function is represented as the following equation,

$$\min_{\mathbf{A}_1[k; f_1], \dots, \mathbf{A}_D[k; f_D]} J^{(AsynDSF-MAX)} = \sum_{\tau=1}^T \max_d z_d[\tau, k]$$

wherein T is the number of the time frames.

7. The method of claim 6, wherein performing the optimization algorithm comprises:

computing a sub-gradient of the objective function as per the following equation,

$$\frac{\partial J^{(\text{AsyncDSF-MAX})}}{\partial \mathbf{a}_{n,d}^*[k; f_d]} = \sum_{\tau=1}^T \frac{\partial \max(z_1[\tau, k], \dots, z_D[\tau, k])}{\partial z_d[\tau, k]} \frac{\partial z_d[\tau, k]}{\partial \mathbf{a}_{n,d}^*[k; f_d]}$$

where

$$\frac{\partial \max(z_1[\tau, k], \dots, z_D[\tau, k])}{\partial z_d[\tau, k]} = \begin{cases} 1, & z_d[\tau, k] = \max(z_1[\tau, k], \dots, z_D[\tau, k]) \\ 0, & \text{otherwise} \end{cases}$$

$$\frac{\partial z_d[\tau, k]}{\partial \mathbf{a}_{n,d}^*[k; f_d]} = \left(\frac{\mathbf{g}_{n,d}[\tau, k]}{\|\mathbf{a}_{n,d}[k; f_d]\|_2} - \frac{\mathbf{a}_{n,d}[k; f_d] \Re\{\mathbf{a}_{n,d}^H[k; f_d] \mathbf{g}_{n,d}[\tau, k]\}}{\|\mathbf{a}_{n,d}[k; f_d]\|_2^3} \right)$$

$$\mathbf{g}_{n,d}[\tau, k] = 2 \frac{\mathbf{x}_d[\tau, k; f_d] (\mathbf{x}_d^H[\tau, k; f_d] \mathbf{a}_{n,d}[k; f_d]) h_{n,d}[\tau, k]}{\|\mathbf{x}_d[\tau, k; f_d]\|_2^2 \|\mathbf{a}_{n,d}[k; f_d]\|_2}$$

$$h_{n,d}[\tau, k] = -\frac{1}{N} \left(\frac{z_d[\tau, k]}{1 - \frac{|\mathbf{a}_{n,d}^H[k; f_d] \mathbf{x}_d[\tau, k; f_d]|^2}{\|\mathbf{a}_{n,d}[k; f_d]\|_2^2 \|\mathbf{x}_d[\tau, k; f_d]\|_2^2}} \right)^{1-r}$$

; and

subtracting a product of the sub-gradient and a learning rate to the mixing matrix plus an optional momentum term to update the mixing matrix

, or

subtracting a product of the sub-gradient and an approximation of the Hessian matrices from the mixing matrix.

8. The method of claim 1, wherein the step of computing the original signals of the signal sources according to the mixing matrix and the vectors $\mathbf{x}_d[\tau, k; f_d]$ of the devices comprises:

constituting a vector $\mathbf{x}[\tau, k] = [\mathbf{x}_1[\tau, k; f_1]^T, \dots, \mathbf{x}_D[\tau, k; f_D]^T]^T \in \mathbb{C}^C$ where $C = \sum c_d$ and D denotes a number of the devices; and

multiplying an inverse of mixing matrix by the vector $\mathbf{x}[\tau, k]$ to obtain the original signals, or

multiplying a time-frequency mask derived from the mixing matrix and $\mathbf{x}[\tau, k]$ by the vector $\mathbf{x}[\tau, k]$ to obtain the original signals.

9. An audio processing system comprising:

a plurality of devices, wherein each of the devices comprises a plurality of microphones, wherein each of the microphones of each of the devices is configured to sense an audio signal; and

a server, wherein the devices and the server are configured to perform steps of:

(a) obtaining the audio signal sensed by each of the microphones of each of the devices, dividing each of the audio signals into a plurality of time frames, and performing time to frequency transform on each of the time frames to generate a vector $\mathbf{x}_d[\tau, k; f_d]$ in a time-frequency (TF) domain, wherein d denotes a d^{th} device, f_d denotes a sampling rate of the d^{th} device, τ denotes a τ^{th} time frame, and k denotes a k^{th} frequency bin index;

(b) setting a mixing matrix $\mathbf{A}[\mathbf{k}] = [\mathbf{A}_1[k; f_1]^T, \dots, \mathbf{A}_D[k; f_D]^T]^T$, wherein D denotes a number of the devices, a matrix $\mathbf{A}_d[k; f_d]$ corresponds to the d^{th} device, a size of the mixing matrix $\mathbf{A}_d$ is $c_d \times N$, c_d is a number of the

microphones of the d^{th} device, and N is a number of a plurality of signal sources;

(c) for each of the signal sources, each of the devices, and each of the time frames, computing a measure of dissimilarity between the vector $x_d[\tau, k; f_d]$ and a vector $a_{n,d}[k; f_d]$ which denotes a n^{th} column of the matrix $A_d[k; f_d]$, wherein n is a positive integer less than or equal to N ;

(d) establishing an objective function according to the measures of dissimilarity, and performing an optimization algorithm according to the objective function to compute the mixing matrix; and

(e) computing a plurality of original signals of the signal sources according to the mixing matrix and the vectors $x_d[\tau, k; f_d]$ of the devices without estimating a sampling rate mismatch between the devices.

10. The audio processing system of claim 9, wherein the devices are configured to transmit the audio signals to the server which is configured to perform the steps (a) to (e).

11. The audio processing system of claim 9, wherein the devices are configured to perform the steps (a) to (c) to calculate a plurality of discriminative terms and transmit the discriminative terms to the server,

wherein the server is configured to compute at least one frame index according to the discriminative terms, and transmit the at least one frame index to the devices to update the mixing matrix.

12. The audio processing system of claim 9, wherein the measure of dissimilarity is a cosine similarity between the vector $\mathbf{x}_d[\tau, k; f_d]$ and the vector $\mathbf{a}_{n,d}[k; f_d]$.

13. The audio processing system of claim 12, wherein the steps further comprise:

computing a discriminative term based on the measure of dissimilarity, wherein the discriminative term is represented as a following equation,

$$z_d[\tau, k] = \left[\frac{1}{N} \sum_{n=1}^N \left[\left(1 - \frac{|\mathbf{a}_{n,d}^H[k; f_d] \mathbf{x}_d[\tau, k; f_d]|^2}{\|\mathbf{a}_{n,d}[k; f_d]\|_2^2 \|\mathbf{x}_d[\tau, k; f_d]\|_2^2} \right) \right]^r \right]^{1/r}$$

wherein r is a real number, and $\mathbf{a}_{n,d}^H[k; f_d]$ is a conjugate transpose of the vector $\mathbf{a}_{n,d}[k; f_d]$.

14. The audio processing system of claim 13, wherein the objective function is represented as a following equation,

$$\min_{\mathbf{A}_1[k; f_1], \dots, \mathbf{A}_D[k; f_D]} J^{(\text{AsyncDSF-MIN})} = \sum_{\tau=1}^T \min_d z_d[\tau, k]$$

wherein T is the number of the time frames.

15. The audio processing system of claim 14, wherein performing the optimization algorithm comprises:

computing a sub-gradient of the objective function as per following equation,

$$\frac{\partial J^{(\text{AsyncDSF-MIN})}}{\partial \mathbf{a}_{n,d}^*[k; f_d]} = \sum_{\tau=1}^T \frac{\partial \min(z_1[\tau, k], \dots, z_D[\tau, k])}{\partial z_d[\tau, k]} \frac{\partial z_d[\tau, k]}{\partial \mathbf{a}_{n,d}^*[k; f_d]}$$

where

$$\frac{\partial \min(z_1[\tau, k], \dots, z_D[\tau, k])}{\partial z_d[\tau, k]} = \begin{cases} 1, & z_d[\tau, k] = \min(z_1[\tau, k], \dots, z_D[\tau, k]) \\ 0, & \text{otherwise} \end{cases}$$

$$\frac{\partial z_d[\tau, k]}{\partial \mathbf{a}_{n,d}^*[k; f_d]} = \left(\frac{\mathbf{g}_{n,d}[\tau, k]}{\|\mathbf{a}_{n,d}[k; f_d]\|_2} - \frac{\mathbf{a}_{n,d}[k; f_d] \Re\{\mathbf{a}_{n,d}^H[k; f_d] \mathbf{g}_{n,d}[\tau, k]\}}{\|\mathbf{a}_{n,d}[k; f_d]\|_2^3} \right)$$

$$\mathbf{g}_{n,d}[\tau, k] = 2 \frac{\mathbf{x}_d[\tau, k; f_d] (\mathbf{x}_d^H[\tau, k; f_d] \mathbf{a}_{n,d}[k; f_d]) h_{n,d}[\tau, k]}{\|\mathbf{x}_d[\tau, k; f_d]\|_2^2 \|\mathbf{a}_{n,d}[k; f_d]\|_2}$$

$$h_{n,d}[\tau, k] = -\frac{1}{N} \left(\frac{z_d[\tau, k]}{1 - \frac{|\mathbf{a}_{n,d}^H[k; f_d] \mathbf{x}_d[\tau, k; f_d]|^2}{\|\mathbf{a}_{n,d}[k; f_d]\|_2^2 \|\mathbf{x}_d[\tau, k; f_d]\|_2^2}} \right)^{1-r}$$

; and

subtracting a product of the sub-gradient and a learning rate to the mixing matrix plus an optional momentum term to update the mixing matrix

, or

subtracting a product of the sub-gradient and an approximation of the Hessian matrices from the mixing matrix.

16. The audio processing system of claim 13, wherein the objective function is represented as a following equation,

$$\min_{\mathbf{A}_1[k; f_1], \dots, \mathbf{A}_D[k; f_D]} J^{(\text{AsyncDSF-MAX})} = \sum_{\tau=1}^T \max_d z_d[\tau, k]$$

wherein T is a number of the time frames.

17. The audio processing system of claim 16, wherein performing the optimization algorithm comprises:

computing a sub-gradient of the objective function as a following equation,

$$\frac{\partial J^{(\text{AsyncDSF-MAX})}}{\partial \mathbf{a}_{n,d}^*[k; f_d]} = \sum_{\tau=1}^T \frac{\partial \max(z_1[\tau, k], \dots, z_D[\tau, k])}{\partial z_d[\tau, k]} \frac{\partial z_d[\tau, k]}{\partial \mathbf{a}_{n,d}^*[k; f_d]}$$

where

$$\frac{\partial \max(z_1[\tau, k], \dots, z_D[\tau, k])}{\partial z_d[\tau, k]} = \begin{cases} 1, & z_d[\tau, k] = \max(z_1[\tau, k], \dots, z_D[\tau, k]) \\ 0, & \text{otherwise} \end{cases}$$

$$\frac{\partial z_d[\tau, k]}{\partial \mathbf{a}_{n,d}^*[k; f_d]} = \left(\frac{\mathbf{g}_{n,d}[\tau, k]}{\|\mathbf{a}_{n,d}[k; f_d]\|_2} - \frac{\mathbf{a}_{n,d}[k; f_d] \Re\{\mathbf{a}_{n,d}^H[k; f_d] \mathbf{g}_{n,d}[\tau, k]\}}{\|\mathbf{a}_{n,d}[k; f_d]\|_2^3} \right)$$

$$\mathbf{g}_{n,d}[\tau, k] = 2 \frac{\mathbf{x}_d[\tau, k; f_d] (\mathbf{x}_d^H[\tau, k; f_d] \mathbf{a}_{n,d}[k; f_d]) h_{n,d}[\tau, k]}{\|\mathbf{x}_d[\tau, k; f_d]\|_2^2 \|\mathbf{a}_{n,d}[k; f_d]\|_2}$$

$$h_{n,d}[\tau, k] = -\frac{1}{N} \left(\frac{z_d[\tau, k]}{1 - \frac{|\mathbf{a}_{n,d}^H[k; f_d] \mathbf{x}_d[\tau, k; f_d]|^2}{\|\mathbf{a}_{n,d}[k; f_d]\|_2^2 \|\mathbf{x}_d[\tau, k; f_d]\|_2^2}} \right)^{1-r}$$

; and

subtracting a product of the sub-gradient and a learning rate to the mixing matrix plus an optional momentum term to update the mixing matrix

, or

subtracting a product of the sub-gradient and an approximation of the Hessian matrices from the mixing matrix.

18. The audio processing system of claim 9, wherein the step of computing the original signals of the signal sources according to the mixing matrix and the vectors $x_d[\tau, k; f_d]$ of the device comprise:

constituting a vector $x[\tau, k] = [x_1[\tau, k; f_1]^T, \dots, x_D[\tau, k; f_D]^T]^T \in \mathbb{C}^C$ where $C = \sum c_d$ and D denotes a number of the devices; and

multiplying an inverse of mixing matrix by the vector $x[\tau, k]$ to obtain the original signals, or

multiplying a time-frequency mask derived from the mixing matrix and $x[\tau, k]$ by the vector $x[\tau, k]$ to obtain the original signals.