A METHOD OF ICA IN TIME-FREQUENCY DOMAIN

Shiro Ikeda

PRESTO, JST
Hirosawa 2-1, Wako, 351-0198, Japan
Shiro.Ikeda@brain.riken.go.jp

Noboru Murata

RIKEN BSI
Hirosawa 2-1, Wako, 351-0198, Japan
Noboru.Murata@brain.riken.go.jp

ABSTRACT

We propose a method of ICA for separating convolutive mixtures of acoustic signals. The acoustic signals recorded in a real environment are not instantaneous but convolutive mixtures, because of the delay and the reflections. In order to separate these signals, it is effective to transform the signals into time-frequency domain. The difficult point in these approaches is the ambiguity of the permutation and amplitude which is unavoidable in original ICA problem. Since the basic ICA approaches cannot solve these ambiguity, we need another approach to solve them. We employed the envelopes of the signals to solve it, and have developed some algorithms. In this article, we show the outline of our orignal method, and some extensions of it. They are, the on-line version and auditory scene analysis problem.

1. INTRODUCTION

One of the good applications of ICA is separating acoustic signals recorded in a real environment. This problem is well known as the name, "cocktail party problem". What is difficult for these problems is that the signals include delays and reflections(Fig.1).

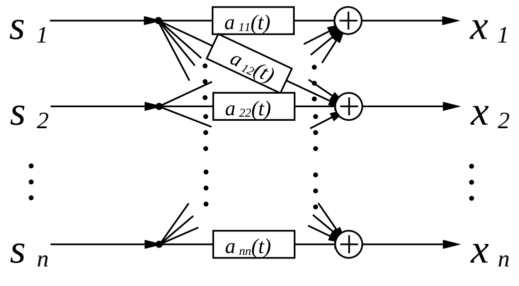

Figure 1: The problem: Convolutive Mixtures

In order to solve this problem, there are some approaches, and one of them is to use decorrelation algorithm [3]. In the algorithm, they approximate the process from the sources to the microphones with FIR filters, and make the inverse filters of those processes to separate the sources. But those algorithms usually takes a lot of times for the calculation, and since they try to use the inverse filters which is also approximated with FIR filters, the impulse response of those inverse filters are usually long and resulting signals are distorted.

In this article, we propose a method based on the windowed-Fourier transform, which is known as the name of spectrogram. If the effect of the delays and the reflections are not too long, we can ignore these convolutions by applying the windowed-Fourier transform. But if we use the windowed-Fourier transform new problems occur, in most of the ICA approaches, they usually ignore the ambiguities of the amplitude and the permutation. We have to remove these ambiguities in order to reconstruct the signals. Our idea is to use the inverse of the decorrelating matrices and the envelope of the speech signal. This is possible because of the temporal structure of the acoustic signals that it is stationary for a short period but not stationary for a long term[5]. We use this time structure to build an algorithm. There are also some other approaches based on the time-frequency domain [6, 10].

We show the basic idea of the algorithm in Section 2, and show some variations in Section 3. We show two variations, one is the on-line algorithm, and the other is separating the single channel recorded sound in which two source are recorded. This problem is equivalent to the one which is dealt in the auditory scene analysis.

2. PROPOSED METHOD

First, we give a formulation of the problem. Source signals are denoted by a vector

$$\boldsymbol{s}(t) = (s_1(t), \cdots, s_n(t))^T, \quad t = 0, 1, 2, \dots. \quad (1)$$

We assume that each component of $\boldsymbol{s}(t)$ is independent of each other and zero mean. When the signals are recorded in a real environment, the observations can be approximated with convolutive mixtures of source signals,

$$\begin{aligned} \boldsymbol{x}(t) = A * \boldsymbol{s}(t) = \left(\sum_k a_{ik} * s_k(t) \right), \\ a_{ik} * s_k(t) = \sum_{\tau=0}^{\tau_{max}} a_{ik}(\tau) s_k(t-\tau), \end{aligned} \tag{2}$$

where $A(t)$ is a function of time, $a_{ik}(t)$ is the impulse response of the process from source signal k to sensor i, and $a_{ik} * s_k(t)$ is the convolution of $a_{ik}(t)$ and $s_k(t)$. The goal of ICA is to separate signals into the components which are mutually independent without knowing operator A and source signals $\boldsymbol{s}(t)$.

If we apply Fourier transform, (2) can be written as,

$$\hat{\boldsymbol{x}}(\omega) = \hat{A}(\omega) \hat{\boldsymbol{s}}(\omega) \tag{3}$$

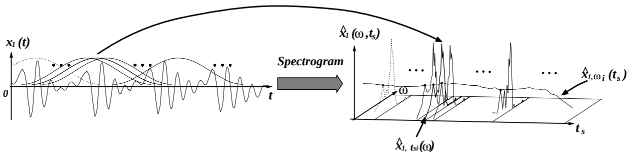

Figure 2: The windowed-Fourier transform

where, $\hat{\boldsymbol{x}}(\omega)$, $\hat{A}(\omega)$ and $\hat{\boldsymbol{s}}(\omega)$ are Fourier transform of $\boldsymbol{x}(t)$, $A(t)$ and $\boldsymbol{s}(t)$ respectively. It is said that the human voice is stationary for a period shorter than a few 10msecs[5]. If it is longer than a few 10msecs and around 100msec, the frequency components of the speech will change its structure, and it is not stationary. From this fact, if the delay and the reflections are not too long, we can approximate the relationship between the sources and the signals with the windowed-Fourier transform as,

$$\hat{\boldsymbol{x}}(\omega, t_s) = \hat{A}(\omega) \hat{\boldsymbol{s}}(\omega, t_s), \tag{4}$$

where $\hat{\boldsymbol{x}}(\omega, t_s)$ and $\hat{\boldsymbol{s}}(\omega, t_s)$ are the windowed-Fourier transform of $\boldsymbol{x}(t)$ and $\boldsymbol{s}(t)$. The windowed-Fourier transform is defined as,

$$\begin{aligned} \hat{\boldsymbol{x}}(\omega, t_s) = \sum_t e^{-j\omega t} \boldsymbol{x}(t) w(t - t_s), \\ \omega = 0, \tfrac{1}{N}2\pi, \ldots, \tfrac{N-1}{N}2\pi, \quad t_s = 0, \mathit{\Delta T}, 2\mathit{\Delta T}, \ldots \end{aligned} \tag{5}$$

where ω denotes the frequency and N denotes the number of points of the discrete Fourier transform, t_s denotes the window position, w is a window function (we used Hamming window) and $\mathit{\Delta T}$ is the shifting interval of moving windows. Let us redefine $\hat{\boldsymbol{x}}(\omega, t_s)$ and $\hat{\boldsymbol{s}}(\omega, t_s)$ for a fixed frequency ω as $\hat{\boldsymbol{x}}_\omega(t_s)$ and ω as $\hat{\boldsymbol{s}}_\omega(t_s)$(see Fig.2). Equation (4) can be rewritten as $\hat{\boldsymbol{x}}_\omega(t_s) = \hat{A}(\omega) \hat{\boldsymbol{s}}_\omega(t_s)$ and it shows that convolutive mixtures are simply an instantaneous mixture for a fixed ω. Therefore we can apply any ICA algorithm for each frequency and separate the signals. As the result, we have a separated time sequence for each frequency,

$$\hat{\boldsymbol{u}}_\omega(t_s) = B(\omega) \hat{\boldsymbol{x}}_\omega(t_s).$$

It seems natural that we can reconstruct the separated signals by aligning these $\hat{\boldsymbol{u}}_\omega(t_s)$ obtained for each frequency and apply the inverse Fourier transform. However, two problems arise. Since ICA algorithms cannot solve the ambiguity of amplitude and permutation, even if we put each component of $\hat{\boldsymbol{u}}_\omega(t_s)$ along with ω, amplitudes are irregular for each frequency and different independent sources will be mixed up. We show how to solve those two problems in the following two subsections.

2.1. Removing the ambiguity of amplitude

The problem of irregular amplitude can be solved by putting back the separated independent components to the sensor input with the inverse matrices $B(\omega)^{-1}$. Let us define $\hat{\boldsymbol{v}}_\omega(t_s; i)$ as,

$$\begin{aligned} \hat{\boldsymbol{v}}_\omega(t_s; i) = B(\omega)^{-1}(0 \ldots 0, \hat{u}_{i,\omega}(t_s), 0 \ldots 0)^T, \\ i = 1, \ldots, n \end{aligned}$$

where $\hat{v}_{k,\omega}(t_s; i)$ represents the input of the i-th independent component of $\hat{\boldsymbol{u}}_\omega(t_s)$ into the k-th ($k = 1, \ldots, n$) sensor. We applied $B(\omega)$ and $B(\omega)^{-1}$ to obtain $\hat{\boldsymbol{v}}_\omega(t_s; i)$, therefore $\hat{\boldsymbol{v}}_\omega(t_s; i)$ has no ambiguity of amplitude. $\hat{\boldsymbol{v}}_\omega(t_s; i)$ is an n dimensional vector, and i changes from i to n, therefore, we have $n \times n$ signals. $\hat{v}_{\omega,j}(t_s; i)$ correspond to estimated input of the i-th source in the j-th sensor. In our experiment, we use two sources and two microphone, and the result of the each experiment has four outputs.

2.2. Removing the ambiguity of permutation

Remaining problem is permutation indeterminateness. We assumed that even for different frequencies, if the original source is the same, the envelopes are similar, and utilize this idea for solving the permutation. We define the envelope with an operator $\mathcal{E}$ as,

$$\mathcal{E}\hat{\boldsymbol{v}}_\omega(t_s; i) = \frac{1}{2M} \sum_{t'_s = t_s - M}^{t_s + M} \sum_{k=1}^{n} |\hat{v}_{k,\omega}(t'_s; i)|, \tag{6}$$

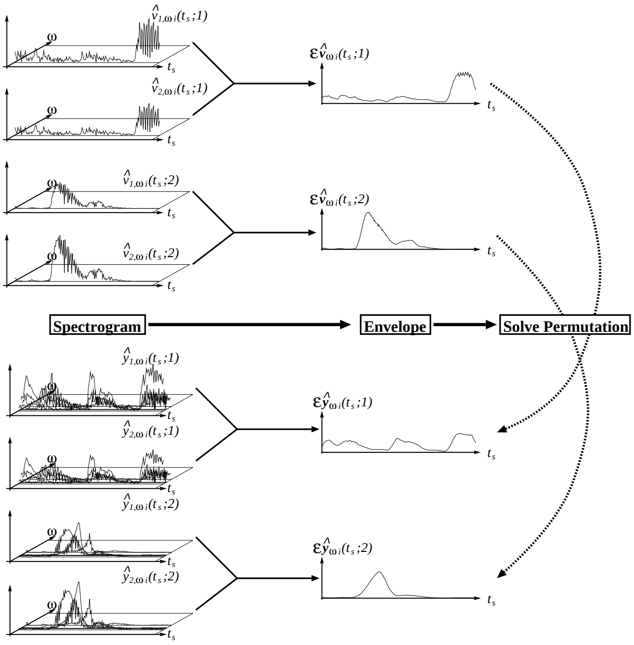
Spectrogram
Envelope
Solve Permutation
$\hat{v}_{1,\omega_i}(t_s;1)$
$\hat{v}_{2,\omega_i}(t_s;1)$
$\hat{v}_{1,\omega_i}(t_s;2)$
$\hat{v}_{2,\omega_i}(t_s;2)$
$\mathcal{E}\hat{v}_{\omega_i}(t_s;1)$
$\mathcal{E}\hat{v}_{\omega_i}(t_s;2)$
$\hat{y}_{1,\omega_i}(t_s;1)$
$\hat{y}_{2,\omega_i}(t_s;1)$
$\hat{y}_{1,\omega_i}(t_s;2)$
$\hat{y}_{2,\omega_i}(t_s;2)$
$\mathcal{E}\hat{y}_{\omega_i}(t_s;1)$
$\mathcal{E}\hat{y}_{\omega_i}(t_s;2)$
ω
t_s

Figure 3: Solving the permutation ambiguity

where M is a positive constant and $\hat{v}_{k,\omega}(t_s;i)$ denotes the k-th element of $\hat{\boldsymbol{v}}_\omega(t_s;i)$. And we defined the similarity among all the envelopes in the same frequencies by,

$$\mathrm{sim}(\omega) = \sum_{i \neq k} \frac{\mathcal{E}\hat{\boldsymbol{v}}_\omega(i) \cdot \mathcal{E}\hat{\boldsymbol{v}}_\omega(k)}{\|\mathcal{E}\hat{\boldsymbol{v}}_\omega(i)\| \|\mathcal{E}\hat{\boldsymbol{v}}_\omega(k)\|}. \tag{7}$$

where the inner product and norm are defined as

$$\mathcal{E}\hat{\boldsymbol{v}}_\omega(i) \cdot \mathcal{E}\hat{\boldsymbol{v}}_{\omega'}(k) = \sum_{t_s} \mathcal{E}\hat{\boldsymbol{v}}_\omega(t_s;i)\mathcal{E}\hat{\boldsymbol{v}}_{\omega'}(t_s;k), \tag{8}$$

$$\|\mathcal{E}\boldsymbol{v}_\omega(i)\| = \sqrt{\mathcal{E}\hat{\boldsymbol{v}}_\omega(i) \cdot \mathcal{E}\hat{\boldsymbol{v}}_\omega(i)}, \tag{9}$$

Based on the similarities of independent components in different frequencies measured with these operations, components are properly classified and the permutation is solved. The procedure is, to find a permutation $\sigma_\omega(i)$ which maximizes correlation between $\mathcal{E}\hat{\boldsymbol{v}}_\omega(t_s;\sigma_\omega(i))$ and $\mathcal{E}\hat{\boldsymbol{y}}(t_s;i) = \mathcal{E}\sum_{\omega'}\hat{\boldsymbol{v}}_{\omega'}(t_s;\sigma_{\omega'}(i))$ inductively (see Figure 3). For the details, see [4]. As a result of solving the permutation, we obtain separated spectrograms as $\hat{\boldsymbol{y}}_\omega(t_s;i)$. Applying inverse Fourier transform, finally we get a set of separated sources

$$\boldsymbol{y}(t;i) = \frac{1}{2\pi} \cdot \frac{1}{W(t)} \sum_{t_s} \sum_{\omega} e^{j\omega(t-t_s)} \hat{\boldsymbol{y}}_\omega(t_s;i), \qquad i = 1, \ldots, n$$

where $W(t) = \sum_{t_s} w(t-t_s)$. Note that each $y_k(t;i)$ represents a separated independent component i on sensor k, and $\sum_i \boldsymbol{y}(t;i) = \boldsymbol{x}(t)$ holds. And finally we obtain $n \times n$ signals from n dimensional inputs.

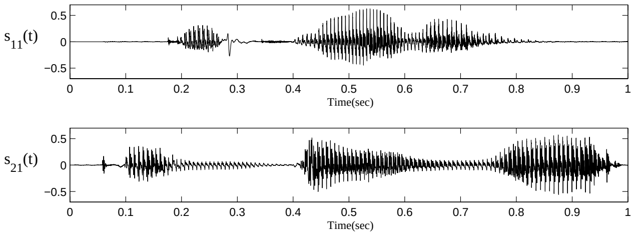

Figure 4: The source signals: each signal was spoken by a different male and recorded with sampling rate of 16kHz. $s_1(t)$ is a word of "good morning" and $s_2(t)$ is a Japanese word "konbanwa".

It is natural to ask a practical question, "Is it really necessary to solve the permutation?". In order to answer this question, we show an example.

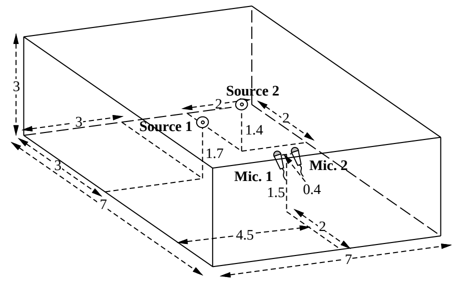

Figure 5: Virtual room for making convolutive mixtures: unit for the length is meter. The strength of the reflection is 0.1 in power for any frequency.

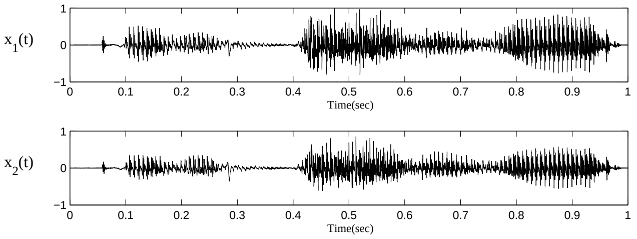

Figure 6: The inputs: Convolutive Mixtures

Figure 4 is the sources recorded separately. We have built a virtual room (Fig.5) in a computer and put microphones and sources in the room. Then mixed the signals on the computer. These mixed signals are shown in Fig.6. These signals are convolutive mixtures.

We separated these mixtures with the proposed algorithm. We applied the windowed-Fourier transform, and applied the separating algorithm (in this case we used the method proposed by Molgedey and Schuster[7]).

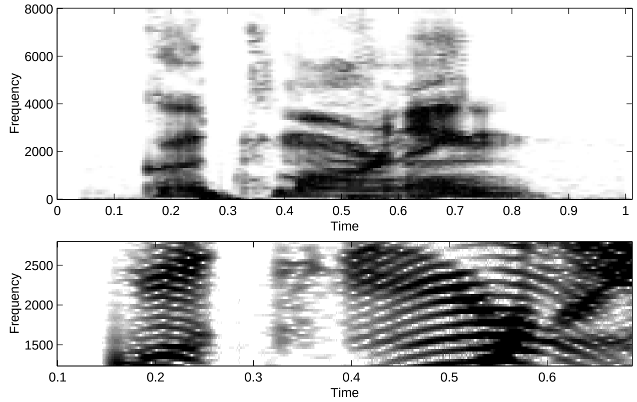

Figure 7: The spectrogram of sources

Figure 7 shows the spectrogram of one of the sources in Fig.4. On the top of the figure, whole spectrogram is shown and a part of it is zoomed up in the bottom.

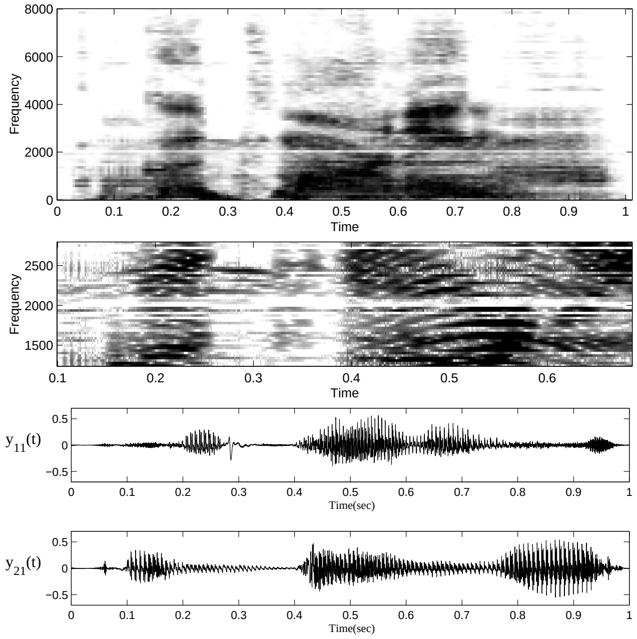

Figure 8: The separated signals including permutation errors

First, we applied our algorithm without solving the permutation ambiguity. One of the resulting spectrogram and the signals are shown in Fig.8. It is clear in the zoomed up figure that there are some discontinuity. In Fig.9, we show the spectrogram and the signals when the permutation ambiguity was solved. The spectrogram is more natural and smoother. From these results, it is clear that the permutation problem can occur, and we have to deal with it carefully.

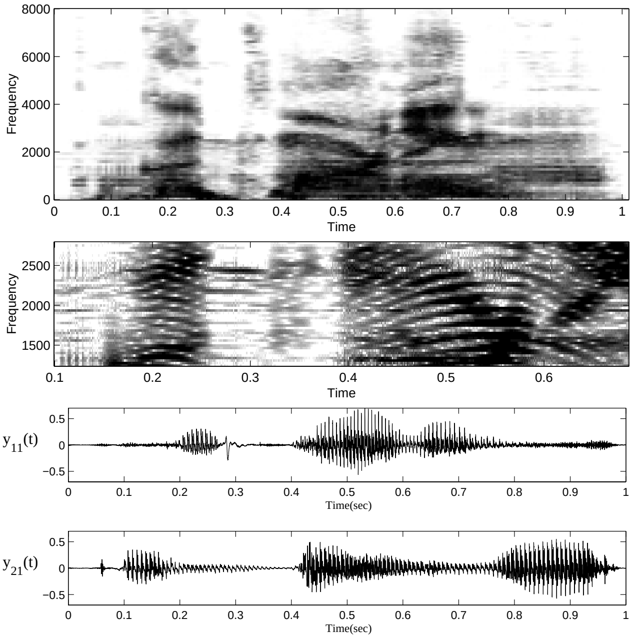

Figure 9: The separated signals

3. VARIATIONS

3.1. Online algorithm

For extracting independent components from the mixed signals in each frequency channel, we use a recurrent neural network architecture [7, 2], in which the output vector is described as

$$\hat{\boldsymbol{u}}(\omega,t_s)=\hat{\boldsymbol{x}}(\omega,t_s)-B(\omega,t_s)\hat{\boldsymbol{u}}(\omega,t_s),$$

where $B(\omega,t_s)$ is a matrix, whose ij element is a connection from the j-th component of output $\hat{\boldsymbol{u}}(\omega,t_s)$ to the i-th component of input $\hat{\boldsymbol{x}}(\omega,t_s)$ and whose diagonal elements are fixed to 0, that means there is no self-recurrent connection in the network. Since $\hat{\boldsymbol{u}}(\omega,t_s)=(B(\omega,t_s)+I)^{-1}\hat{\boldsymbol{x}}(\omega,t_s)$, the source signals are completely extracted when $A(\omega)=I+B(\omega,t_s)$, where I is the identity matrix.

In the experiment described below, we adopt the following learning rule (see [1] for derivation of the algorithm and its stability analysis),

$$\begin{aligned}
&B(\omega, t_s + \mathit{\Delta} T) = B(\omega, t_s) \\
&\quad - \eta \left(B(\omega, t_s) + I\right) \left(\operatorname{diag}\left(\phi(\boldsymbol{z})\boldsymbol{z}^*\right) - \phi(\boldsymbol{z})\boldsymbol{z}^*\right), \\
&\quad \boldsymbol{z} = \hat{\boldsymbol{u}}(\omega, t_s)
\end{aligned} \tag{10}$$

where diag(·) makes a diagonal matrix with the diagonal elements of its argument, * denotes complex conjugate, and

$$\phi(z) = \tanh(\mathrm{Re}(z)) + i \cdot \tanh(\mathrm{Im}(z)) \tag{11}$$

which operates component-wise to a column vector [10]. With using estimated matrix $B(\omega, t_s) + I$ and one independent component we obtain separated independent components of observation in each frequency as

$$\hat{\boldsymbol{v}}_\omega(t_s; i) = (B(\omega, t_s) + I)(0, \ldots, \hat{u}_i(\omega, t_s), \ldots, 0)^T. \tag{12}$$

As described in the last section, because of inherent indeterminacy of ICA problem, correspondence of $\hat{\boldsymbol{v}}_\omega(t_s; i)$ with another frequency is ambiguous. In our approach, individually separated frequency components are combined again based on the common temporal structure of original source signals. The procedure is the same as what we used in the last section. For more detailed explanation about the practical implementation, see [4, 8, 9]. We heard the results, and they are separated clearly.

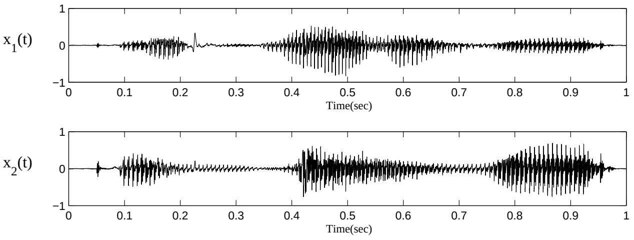

Figure 10: Inputs

3.2. Separating single input

We show another variation of our approach. The problem in this subsection is to separate a single channel input which include two sources. This is equivalent to what is dealt with in the auditory scene analysis. In this experiment the input $x(t)$ which is shown in Fig.12 is mixed on the computer using the signals in Fig.5.

Basically, we have only one input, and it seems to be impossible to separate it into two signals. Theoretically, it is true, but we make an assumption here that,

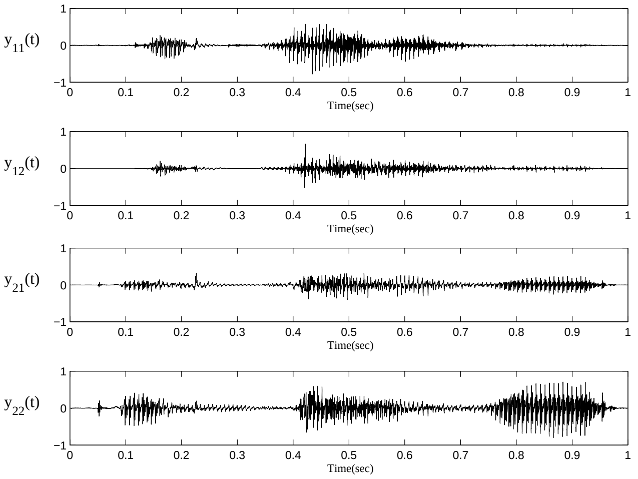

Figure 11: Outputs of the online algorithm

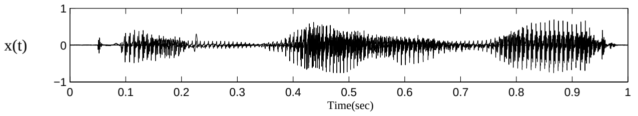

Figure 12: Inputs: $x(t) = 0.5s_1(t) + 0.5s_2(t)$, where, $s_1(t)$ and $s_2(t)$ are the same as those in Fig.5.

$$\begin{pmatrix} x_\omega(t_s) \\ x_{\omega+\delta\omega}(t_s) \end{pmatrix} \simeq A(\omega, t_s) \begin{pmatrix} s_1(\omega, t_s) \\ s_2(\omega, t_s) \end{pmatrix}. \tag{13}$$

This assumption means that the time-frequency component $x_\omega(t_s)$ and another component of its successive frequency $\omega + \delta\omega$ can be approximated with the linear mixture of the source signals' time-frequency component $s_i(\omega, t_s)$. The assumption cannot be true, however from the continuity of the spectrogram in the direction of the successive frequency may support this approximation. Under this assumption, we can apply ICA algorithm.

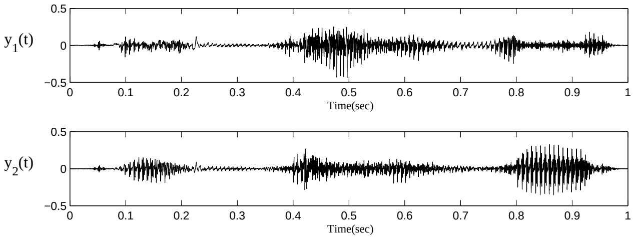

Figure 13: Results of the single input problem

It is said that $A(\omega, t_s)$ is changing its structure with around 100msec. This means, the frequency structures of the signals are changing with around 100msec. In order to solve the separation problem of these signals, we have three possible solutions. One is to use a batch learning algorithm and one of the other is to use an on-line algorithm. Also we have another possibility of using a hybrid of these two algorithms. In this type of approach, batch learning is applied with shifting the time little by little. We used this hybrid approach. The signals was sliced into 0.3sec with the overlap of 0.2sec, and applied batch algorithm shown in section 2. The result is shown in Fig.13. From the graph, it seems that the sources are separated, but there still a great room for improvement.

4. CONCLUSION

It is well-known that humans ears are doing a kind of time-frequency analysis with cochlea. From this evidence, it is plausible to use the time-frequency analysis for sounds in natural environments(such as music and speech). We have applied the ICA algorithm in the time-frequency domain, and showed its possibility. We also showed the inherent problem of ICA which makes a great problem in the case of time-frequency analysis.

Although this time-frequency approach has been pointed out before [6, 10], the permutation problem was not dealt with deeply in their approaches. This is because they are rather minor effect in the practical problems. In the on-line algorithm, they usually use a special parameterization as shown in (10), and in our modified approach of decorrelation algorithm[7, 9], the source are sorted in the order of its power in each frequency. These alignment which is usually not considered deeply, will effect on the permutation problem. But if the environment is rather complex, we have to take the permutation problem into account.

We have shown some variations of the algorithms. For the case of on-line algorithm, we are working on its implementation with a hardware, and for the single input problem, we are working for better parameterization and its implementation. We also think that it is possibility of combining this basic time-frequency approach with other techniques.

5. REFERENCES

[1] Shun-ichi Amari, Tian-Ping Chen, and Andrzej Cichocki. Stability analysis of learning algorithms for blind source separation. *Neural Networks*, 10(8):1345–1351, 1997.

[2] Anthony J. Bell and Terrence J. Sejnowski. An information maximization approach to blind separation and blind deconvolution. *Neural Computation*, 7:1129–1159, 1995.

[3] Scott C. Douglas and Andrzej Cichocki. Neural networks for blind decorrelation of signals. *IEEE Trans. Signal Processing*, 45(11):2829–2842, nov 1997.

[4] Shiro Ikeda and Noboru Murata. An approach to blind source separation of speech signals. In *Proceedings of ICANN'98*, 1998.

[5] Hideki Kawahara and Toshio Irino. Exploring temporal feature representations of speech using neural networks. Technical Report SP88-31, IEICE, Tokyo, 1988. (in Japanese).

[6] Te-Won Lee, Andreas Ziehe, Reinhold Orglmeister, and Terrence Sejnowski. Combining time-delayed decorrelation and ICA: towards solving the cocktail party problem. In *Proceedings of ICASSP'98*, 1998.

[7] L. Molgedey and H. G. Schuster. Separation of a mixture of independent signals using time delayed correlations. *Phys. Rev. Lett.*, 72(23):3634–3637, 1994.

[8] Noboru Murata and Shiro Ikeda. An on-line algorithm for blind source separation on speech signals. In *Proceedings of NOLTA'98*, 1998.

[9] Noboru Murata, Shiro Ikeda, and Ziehe Andreas. An approach to blind source separation based on temporal structure of speech signals. submitted to IEEE trans. on Signal Processing, 1998.

[10] Paris Smaragdis. Blind separation of convolved mixtures in the frequency domain. In *International Workshop on Independence & Artificial Neural Networks*, University of La Laguna, Tenerife, Spain, February 1998.