International Workshop on Acoustic Signal Enhancement 2012, 4-6 September 2012, Aachen

FAST STEREO INDEPENDENT VECTOR ANALYSIS AND ITS IMPLEMENTATION ON MOBILE PHONE

Nobutaka Ono

National Institute of Informatics
2-1-2 Hitotsubashi, Chiyoda-ku, Tokyo, 101-8430, Japan
onono@nii.ac.jp

ABSTRACT

In this paper, a fast algorithm for specifically stereo independent vector analysis (IVA) is presented, which consists of 1) weighted covariance matrix update and 2) demixing matrix update. Unlike the previous work by the author [1], the demixing matrix is updated by directly solving the generalized eigenvalue problem in a closed-form, which is possible only in the two dimension case. A fast computation exploiting vector operation and its implementation on iPhone are also discussed.

Index Terms— independent vector analysis, auxiliary function, generalized eigenvalue problem, mobile phone

1. INTRODUCTION

Blind source separation (BSS) is an effective technique to extract a desired source from mixtures. For convolutive overdetermined mixtures, independent component analysis (ICA) followed by permutation correction [2] and independent vector analysis (IVA) [3, 4, 5] in the frequency domain have been developed as standard techniques in this decade. In the frequency-domain approach, demixing matrices are estimated at each frequency bin, and due to the larger number of frequency bins (typically 2049, 4097, etc.), reducing the computation time is still a major challenge in this field [6]. Especially in the applications of mobile phones or hearing aids, the fast computation is more critical because of the limitations of CPU performance or batteries.

Recently, the author has derived a new effective algorithm for ICA on the basis of an auxiliary function technique [7] and has extended it to IVA [1]. This paper, which focuses on the stereo case, presents a much faster algorithm of IVA. A fast computation exploiting vector operation and its implementation on iPhone are also discussed.

This research was partially supported by Grand-in-Aid for Scientific Research (A) (23240023) from the Ministry of Education, Culture, Sports, Science and Technology of Japan. The author acknowledges Redec co., Ltd. for contribution to development of the presented AuxIVA2 on iPhone.

2. INDEPENDENT VECTOR ANALYSIS

2.1. BSS in Frequency Domain

Assume here that K sources are observed by K microphones and their Short-time Fourier Transform (STFT) representations are obtained. Let $\boldsymbol{s}(\omega,\tau) = (s_1(\omega,\tau) \ \cdots \ s_K(\omega,\tau))^t$ and $\boldsymbol{x}(\omega,\tau) = (x_1(\omega,\tau) \ \cdots \ x_K(\omega,\tau))^t$ be the vector representation of the source signal and the observation signal at (ω,τ)th time-frequency bin, respectively, where ^{t} denotes vector transpose. In frequency-domain approach for convolutive mixture, a linear mixing model in frequency domain:

$$\boldsymbol{x}(\omega,\tau) = A(\omega)\boldsymbol{s}(\omega,\tau) \tag{1}$$

is assumed where $A(\omega)$ is a mixing matrix, and the sources are estimated by a linear demixing process:

$$\boldsymbol{y}(\omega,\tau) = W(\omega)\boldsymbol{x}(\omega,\tau), \tag{2}$$

where $W(\omega) = (\boldsymbol{w}_1(\omega) \ \cdots \ \boldsymbol{w}_K(\omega))^h$ is a demixing matrix, ^{h} denotes Hermitian transpose, and $\boldsymbol{y}(\omega,\tau) = (y_1(\omega,\tau) \ \cdots \ y_K(\omega,\tau))^t$ represents the estimated sources.

2.2. Objective Function of IVA

In IVA, assuming a multivariate p.d.f. for sources to exploit the dependencies over frequency components, the demixing matrices are estimated by minimizing the following objective function, which is derived from Kullback-Leibler divergence between the p.d.f. of the observation and that of the independent source model [3, 4, 5].

$$J(\boldsymbol{W}) = \frac{1}{N_\tau}\sum_{\tau=1}^{N_\tau}\sum_{k=1}^{K} E[G(\boldsymbol{y}_k(\tau))] - \sum_{\omega=1}^{N_\omega}\log|\det W(\omega)| \tag{3}$$

where $\det$ represents determinant of matrix, $\boldsymbol{W}$ denotes a set of $W(\omega)$ $(\omega = 1, \cdots, N_\omega)$, and $\boldsymbol{y}_k(\tau)$ is the source-wise vector representation defined as $\boldsymbol{y}_k(\tau) = (y_k(1,\tau) \ \cdots \ y_k(N_\omega,\tau))^t$ and $G(\boldsymbol{y}_k)$ is called a contrast function and has a relationship $G(\boldsymbol{y}_k) = -\log p(\boldsymbol{y}_k)$ where $p(\boldsymbol{y}_k)$ represents a multivariate p.d.f. of sources. Note that the following discussion is valid when assuming any spherically symmetric and super Gaussian $p(\boldsymbol{y}_k)$ [1, 7].

2.3. Auxiliary-function-based Independent Vector Analysis (AuxIVA)

The fast and convergence-guaranteed algorithm for ICA based on the auxiliary function method was first presented in [7], and it was extended to IVA in [1]. The auxiliary-function-based IVA algorithm consists of the following two alternative updates.

Weighted covariance matrix update: The weighted covariance matrices $V_k(\omega)$ for all ω are updated as follows.

$$r_k(\tau) = \sqrt{\sum_{\omega=1}^{N_\omega} |\boldsymbol{w}_k^h(\omega)\boldsymbol{x}(\omega,\tau)|^2}, \tag{4}$$

$$V_k(\omega) = \frac{1}{N_\tau}\sum_{\tau=1}^{N_\tau}\left[\frac{G'(r_k(\tau))}{r_k(\tau)}\boldsymbol{x}(\omega,\tau)\boldsymbol{x}^h(\omega,\tau)\right]. \tag{5}$$

Demixing matrix update: $\boldsymbol{w}_k(\omega)$ is updated by minimizing the following auxiliary function.

$$Q(W(\omega), \boldsymbol{V}(\omega)) = \frac{1}{2}\sum_{k=1}^{K}\boldsymbol{w}_k^h(\omega)V_k(\omega)\boldsymbol{w}_k(\omega) - \log|\det W(\omega)| + R(\omega), \tag{6}$$

where $\boldsymbol{V}(\omega)$ represents a set of $V_k(\omega)$ for any k and $R(\omega)$ is a constant term independent of $W(\omega)$. $\partial Q/\partial \boldsymbol{w}_k(\omega) = 0$ leads to the following simultaneous vector equations for $k = 1, \cdots, K,\ l = 1, \cdots, K$.

$$\boldsymbol{w}_l^h(\omega)V_k(\omega)\boldsymbol{w}_k(\omega) = \delta_{kl}. \tag{7}$$

Since a general ($K > 2$) closed-form solution for eq. (7) has yet to be discovered [10], sequentially updating each $\boldsymbol{w}_k(\omega)$ on the basis of orthogonal projection while keeping other $\boldsymbol{w}_l$s ($l \neq k$) fixed has been considered [1] so far.

3. FAST CALCULATION FOR STEREO INDEPENDENT VECTOR ANALYSIS

3.1. Demixing Matrix Update Based on Generalized Eigenvectors

When $K = 2$, eq. (7) represents

$$\boldsymbol{w}_1^h V_1 \boldsymbol{w}_1 = 1, \quad \boldsymbol{w}_2^h V_2 \boldsymbol{w}_2 = 1, \tag{8}$$

$$\boldsymbol{w}_2^h V_1 \boldsymbol{w}_1 = 0, \quad \boldsymbol{w}_1^h V_2 \boldsymbol{w}_2 = 0, \tag{9}$$

where we omit the variable ω for simplicity. In this case, they have a closed-form solution [7, 11]. Eq. (9) indicates that both $V_1\boldsymbol{w}_1$ and $V_2\boldsymbol{w}_1$ are orthogonal to $\boldsymbol{w}_2$. Because the direction orthogonal to $\boldsymbol{w}_2$ is uniquely determined in the two-dimensional space, $V_1\boldsymbol{w}_1$ and $V_2\boldsymbol{w}_1$ have to be parallel. In the same way, $V_1\boldsymbol{w}_2$ and $V_2\boldsymbol{w}_2$ are also parallel. Therefore, $\boldsymbol{w}_1$ and $\boldsymbol{w}_2$ are obtained from the general eigenvalue problem as

$$V_2\boldsymbol{e}_k = \lambda_k V_1 \boldsymbol{e}_k \quad (k = 1, 2), \tag{10}$$

where we define $\lambda_1 \geq \lambda_2$. We can here assume that V_1 and V_2 are positive definite since they are weighted covariance matrices. Note the following lemma.

Lemma 1 *λ_1 and λ_2 in eq. (10) are positive real numbers.*

Proof: When $\boldsymbol{e}_k^h$ is multiplied from left to eq. (10), we have $\boldsymbol{e}_k^h V_2 \boldsymbol{e}_k = \lambda_k \boldsymbol{e}_k^h V_1 \boldsymbol{e}_k$. Therefore, $\lambda_k = (\boldsymbol{e}_k^h V_2 \boldsymbol{e}_k)/(\boldsymbol{e}_k^h V_1 \boldsymbol{e}_k)$ for $k = 1, 2$ are positive real numbers for positive definite matrices V_1 and V_2. ■

If eq. (8) is taken into account, two solutions seem to be possible:

$$\boldsymbol{w}_1 = \frac{\boldsymbol{e}_1}{\sqrt{\boldsymbol{e}_1^h V_1 \boldsymbol{e}_1}}, \quad \boldsymbol{w}_2 = \frac{\boldsymbol{e}_2}{\sqrt{\boldsymbol{e}_2^h V_2 \boldsymbol{e}_2}} \tag{11}$$

or

$$\boldsymbol{w}_1 = \frac{\boldsymbol{e}_2}{\sqrt{\boldsymbol{e}_2^h V_1 \boldsymbol{e}_2}}, \quad \boldsymbol{w}_2 = \frac{\boldsymbol{e}_1}{\sqrt{\boldsymbol{e}_1^h V_2 \boldsymbol{e}_1}} \tag{12}$$

However, the solution for minimizing eq. (6) is given by eq. (11) as follows.

Lemma 2 *Eq. (11) makes eq. (6) smaller than eq. (12).*

Proof: Both eq. (11) and eq. (12) give $\boldsymbol{w}_k^h V_k \boldsymbol{w}_k = 1$ ($k = 1, 2$) in eq. (6). Therefore, let us consider only $\det$ term in eq. (6)

$$\begin{aligned}
&\det\left[\frac{\boldsymbol{e}_1}{\sqrt{\boldsymbol{e}_1^h V_1 \boldsymbol{e}_1}}\ \frac{\boldsymbol{e}_2}{\sqrt{\boldsymbol{e}_2^h V_2 \boldsymbol{e}_2}}\right] \\
&= \frac{1}{\sqrt{\boldsymbol{e}_1^h V_1 \boldsymbol{e}_1}} \cdot \frac{1}{\sqrt{\boldsymbol{e}_2^h V_2 \boldsymbol{e}_2}} \cdot \det[\boldsymbol{e}_1 \boldsymbol{e}_2] \\
&= \sqrt{\frac{\lambda_1}{\lambda_2}} \cdot \frac{1}{\sqrt{\boldsymbol{e}_1^h V_2 \boldsymbol{e}_1}} \cdot \frac{1}{\sqrt{\boldsymbol{e}_2^h V_1 \boldsymbol{e}_2}} \cdot \det[\boldsymbol{e}_1 \boldsymbol{e}_2] \\
&= \sqrt{\frac{\lambda_1}{\lambda_2}} \cdot \det\left[\frac{\boldsymbol{e}_2}{\sqrt{\boldsymbol{e}_2^h V_1 \boldsymbol{e}_2}}\ \frac{\boldsymbol{e}_1}{\sqrt{\boldsymbol{e}_1^h V_2 \boldsymbol{e}_1}}\right]
\end{aligned} \tag{13}$$

Since $\lambda_1 \geq \lambda_2$, eq. (11) obviously minimizes the objective function. ■

This lemma means that the permutation can be corrected by gathering eigenvectors corresponding to larger or smaller eigenvalues over frequency bins. Therefore, the new update procedure for the demixing matrix update in AuxIVA, denoted as AuxIVA2, is summarized as follows.

1. Calculate $H = V_1^{-1}V_2$.
2. Find two eigenvectors $\boldsymbol{e}_1$ and $\boldsymbol{e}_2$ of H such that the eigenvalue of $\boldsymbol{e}_1$ is larger than or equal to that of $\boldsymbol{e}_2$.
3. Calculate $\boldsymbol{w}_1$ and $\boldsymbol{w}_2$ by eq. (11).

3.2. Explicit Forms of Generalized Eigenvectors in 2×2 Matrix Case

Since λ_k for $k = 1, 2$ in eq. (10) are the solutions of

$$\lambda^2 - \text{tr}(H)\lambda + \text{det}(H) = 0, \quad (14)$$

where tr denotes the trace of matrix, they are given by

$$\lambda_k = \frac{\text{tr}(H) \pm \sqrt{\text{tr}(H)^2 - 4\text{det}(H)}}{2} \quad (k = 1, 2), \quad (15)$$

where $\sqrt{\ }$ is here defined such that its real part is nonnegative. Then, the larger and smaller eigenvalues are deterministically obtained by the following lemma.

Lemma 3 *Let*

$$\lambda_1 = \frac{tr(H) + \sqrt{tr(H)^2 - 4det(H)}}{2}, \quad (16)$$

$$\lambda_2 = \frac{tr(H) - \sqrt{tr(H)^2 - 4det(H)}}{2}. \quad (17)$$

Then, $\lambda_1 \geq \lambda_2$.

Proof: From Lemma 1, λ_1 and λ_2 are positive real numbers. Thus, both $\lambda_1 + \lambda_2$ and $\lambda_1 - \lambda_2$ are real numbers. This indicates that $\sqrt{\text{tr}(H)^2 - 4\text{det}(H)}$ in eq. (15) is a real number, and hence, it is nonnegative by definition. ■

Then, the two eigenvectors are obtained by

$$\boldsymbol{e}_1 = \begin{pmatrix} H_{22} - \lambda_1 \\ -H_{21} \end{pmatrix}, \quad \boldsymbol{e}_2 = \begin{pmatrix} -H_{12} \\ H_{11} - \lambda_2 \end{pmatrix}, \quad (18)$$

where H_{ij} denotes ijth element of H.

3.3. Fast Computation with Vector Operation

When a coding platform has a fast library for vector arithmetic operation, the calculation of eq. (16), eq. (17), and eq. (18) at each frequency bin can be effectively computed by performing the same type of calculation at each frequency bin as vector operations. For example, in the Matlab case, to calculate the determinant of 2×2 matrices at all frequency bins, a dot operation as

```
d(1,:)=V(1,1,:).*V(2,2,:)-V(1,2,:).*V(2,1,:);
```

is faster than frequency-wise calculation as

```
for f=1:N_f;
 d(1,f)=det(V(:,:,f));
end;
```

for a typical number of N_f (the number of frequency bins) such as 2049, 4097, etc. In the same way, eq. (16), eq. (17), and eq. (18) at every frequency bin can be calculated together exploiting such a vector operation. Note that this computation is possible because the algorithm includes no conditional branch and the convergence is theoretically guaranteed (we do not need to check whether the divergence happens or not).

Finally, we can replace $H = V_1^{-1}V_2$ by $H = \det(V_1) \cdot V_1^{-1}V_2$ for reducing calculation because the scale normalization in eq. (11) follows.

Table 1. Experimental conditions

microphone spacing	2.83cm
source-microphone distance	2m
source direction	$10°$ to $170°$ by $20°$
reverberation time	300ms
signal length	10s
sampling frequency	16kHz
frame length	4096
frame shift	2048
window function	hamming

4. EXPERIMENTAL EVALUATIONS

To evaluate its separation performance and convergence speed, AuxIVA2 was applied for synthesized convolutive mixtures of speech. The impulse responses from nine directions recorded in a variable reverberation room (E2A) from RWCP Sound Scene Database in Real Acoustical Environments [9] were used. Also, we selected nine speech utterances from the ATR Japanese speech database (Set B), assigned them to each of the nine directions, and convoluved each of them after downsampling to 16kHz. We prepared all pair combinations (${}_9C_2 = 36$) of them. Other experimental conditions are summarized in Table 1. The proposed algorithm was compared with the natural gradient-type IVA with different step sizes ($\mu = 0.1, 0.2, 0.3$), and AuxIVA with orthogonal projection (which is here denoted as AuxIVA1) in [1]. All algorithms used $G(\boldsymbol{y}_k) = G_R(r_k) = r_k$ as a contrast function. The initial value of the demixing matrix was given by the identity matrix for simplicity. The estimated sources were calculated by applying the estimated demixing matrix with Projection back [8]. The performance was evaluated by the average of SIR improvement over all sources and trials calculated by BSS toolbox [12] at the first five iterations and every 10 iterations.

Fig. 1 shows resultant SIR improvements. Thanks to the closed-form update, AuxIVA2 shows remarkably faster convergence and better separation performance than others.

The experiments were performed in Matlab ver. 7.13 (R2011b) on a laptop PC with Intel Core i7-2620M 2.70GHz. Actual computational times in this environment are compared in Table 2. AuxIVA2 (freq.) and AuxIVA2 (vect.) represent the two different implementations of in AuxIVA2 using eig() function at each frequency bin (frequency-wise) and calculating all frequency bins together with dot operation (vector operation) for the demixing matrix update, respectively. We confirmed that the two implementations obtain identical results except for a negligible numerical difference (less than 10^{-8} dB in SIR). Table 2 shows that exploiting the vector operation effectively reduces the actual computation time.

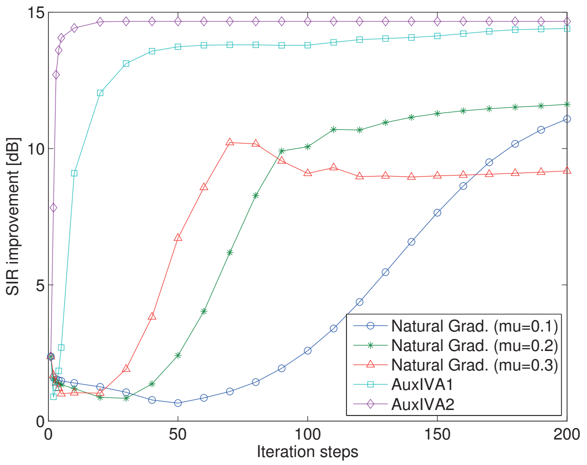

Fig. 1. Averaged SIR improvements at the first five and every 10 iterations by AuxIVA updates and the natural gradient updates with different step sizes

Table 2. Averaged computation time per iteration [s]

Algorithm	Computation time
Natural gradient	0.075
AuxIVA1	0.14
AuxIVA2 (freq.)	0.47
AuxIVA2 (vect.)	0.14

4.1. Implementation on iPhone

As a prototype stereo BSS system on mobile phone, the whole algorithm of AuxIVA2 has been implemented on iPhone in cooperation with Redec co., Ltd. [13]. In this implementation, a vector library named vDSP on Mac OS X [14] was exploited as much as possible for fast computation. In this application, BSS can be applied for stereo signals directly recorded by a stereo microphone connected to iPhone shown in Fig. 2, or stereo signals transferred from a PC. In both cases, the separation is performed in a batch processing manner. The calculation time is almost linear to the input signal length, and under the conditions of 16kHz sampling frequency, 4096 points frame length, half frame shift, and 10 iterations, it took 1.7s to separate 10s input and 23.2s for 120s input on iPhone 4, which means a real time factor (RTF) of less than 1/5 was achieved.

5. CONCLUSIONS

This paper presented a fast algorithm for stereo independent vector analysis. Thanks to the eigenvector-based demixing matrix update, which is possible in only the stereo case, the algorithm achieved a remarkable convergence speed with good separation performance. A fast computation of eigenvectors at each frequency bin while exploiting vector operation was also discussed. The implementation of the whole algorithm on iPhone4 achieved a real time factor (RTF) of less than 1/5.

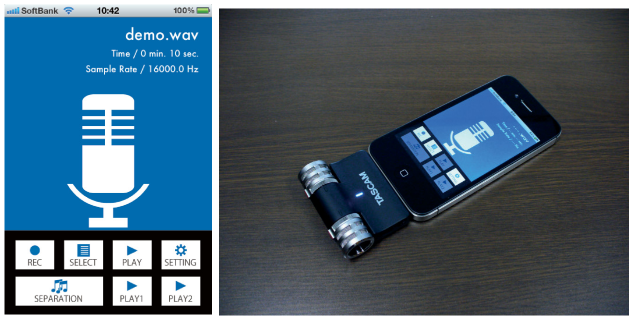

Fig. 2. A screen image of the stereo BSS iPhone application (left) and a photo of iPhone4 with a stereo microphone (TASCAM iM2 provided by TEAC corporation)

6. REFERENCES

[1] N. Ono, "Stable and Fast Update Rules for Independent Vector Analysis Based on Auxiliary Function Technique," *Proc. WASPAA*, pp. 189-192, Oct. 2011.

[2] H. Sawada, R. Mukai, S. Araki, S. Makino, "A Robust and Precise Method for Solving the Permutation Problem of Frequency-Domain Blind Source Separation,"*IEEE Trans. SAP*, vol. 12, no. 5, pp. 530-538, 2004.

[3] A. Hiroe, "Solution of Permutation Problem in Frequency Domain ICA Using Multivariate Probability Density Functions," *Proc. ICA*, pp. 601–608, 2006.

[4] T. Kim, T. Eltoft, and T.-W. Lee, "Independent Vector Analysis: An Extension of ICA to Multivariate Components," *Proc. ICA*, pp. 165–172, 2006.

[5] T. Kim, H. T. Attias, S.-Y. Lee, and T.-W. Lee, "Blind Source Separation Exploiting Higher-order Frequency Dependencies," *IEEE Trans. ASLP*, vol. 15, no. 1, pp. 70–79, 2007.

[6] K. Osako, Y. Mori, Y. Takahashi, H. Saruwatari, and K. Shikano, "Fast Convergence Blind Source Separation based on Frequency Subband Interpolation by Null Beamforming," *Proc. WASPAA*, pp. 42-45, Oct. 2007.

[7] N. Ono and S. Miyabe, "Auxiliary-function-based Independent Component Analysis for Super-Gaussian Sources," *Proc. LVA/ICA*, pp.165-172, 2010.

[8] N. Murata, S. Ikeda, and A. Ziehe, "An Approach to Blind Source Separation Based on Temporal Structure of Speech Signals," *Neurocomputing*, vol. 41, no. 1, pp. 1–24, 2001.

[9] S. Nakamura, K. Hiyane, F. Asano, T. Nishiura, and T. Yamada, "Acoustical Sound Database in Real Environments for Sound Scene Understanding and Hands-Free Speech Recognition," *Proc. LREC*, pp. 965-968, 2000.

[10] A. Yeredor, "On Hybrid Exact-Approximate Joint Diagonalization," *Proc. CAMSAP*, pp. 312–315, 2009.

[11] T. Yoshioka, T. Nakatani, and M. Miyoshi, "An integrated method for blind separation and dereverberation of convolutive audio mixtures," *Proc. EUSIPCO*, Aug. 2008.

[12] E. Vincent, C. Fevotte, and R. Gribonval, "Performance Measurement in Blind Audio Source Separation," *IEEE Trans. ASLP*, vol. 14, no. 4, pp. 1462–1469, 2006.

[13] http://www.redec.co.jp/eng/index.htm

[14] https://developer.apple.com/library/mac/#documentation/Accelerate/Reference/vDSPRef/Reference/reference.html