New Image Encryption and Compression Method Based on Independent Component Analysis

Masanori Ito
and Noboru Ohnishi
Graduate School of Information Science
Nagoya University
Furo-cho, Chikusa-ku, Nagoya, 464–8603 Japan
Email: ito-m@nagoya-u.jp

Ayman Alfalou
ISEN-Brest
20 rue de cuirassé Bretagne
29228 Brest Cedex 2, France

Ali Mansour
ENSIETA
2 Rue François Verny
29806 Brest, France

Abstract—**For telecommunication systems, various compression and encryption techniques are proposed to satisfy a fast and secure transmission. However these two techniques have been studied separately. In this paper we propose a method combining encryption and compression based on Independent Component Analysis (ICA) and Discrete Cosine Transform (DCT).**

For encryption, target images are covered with an insignificant image to hide them and their mixtures to be transmitted are obtained. The receiver reconstructs the original images applying some ICA algorithm to the mixed images. In compression process, using DCT and simple low pass filter, the size of transmission can be reduced.

Throughout several computer simulations, the performance of the proposed method is confirmed.

I. Introduction

Telecommunication becomes one of our modern societies characteristics, which requires more and more new techniques to meet the increasing needs of a modern society. However, for any communication system, it is necessary to take into account two major requirements: a fast transmission to send the information from a transmitter to a receiver (that can be done using an efficient compression technique) and a secure transmission of information which can be achieved using a powerful encoding algorithm. To satisfy these constraints, new compression and encryption methods allowing a fast and secure information transmission are proposed in the literature. However, these methods (compression and encoding) are often developed in an independent manner although they are strongly connected and one influences the other. Accordingly, we propose to combine compression with encryption and propose a new method of compression and encryption at same time. Our new method is based on the multiplexing of information (Fusion) in the spectral field by using a model of Blind Source Separation (BSS) problem in the encoding phase and an Independent Component Analysis (ICA) algorithm in the decoding phase.

Recently, ICA has been greatly developed by various researchers [1]–[4]. ICA was introduced in order to solve the BSS problems which consists of the separation of independent sources using observed mixed signals without a strong knowledge about sources and mixing process. The simplicity and the good performances of these techniques have been a strong motivation for many researchers. Actually, this problem has been applied in various situations, such as: in an airport surveillance [5], the analysis of the brain tumor [6], the encryption technique in order to enhance the security level of optical encryption methods [7].

It is supposed that we transmit important images to a receiver, preventing non-authorized people from intercepting the images. In order to encrypt the images we cover the images with an insignificant image. In addition we would like to compress the transmitting data, to achieve a high-speed communication. For this purpose we utilized Discrete Cosine Transform (DCT) and cut out the higher-frequency components because most of the power are concentrated in the lower frequency bands by DCT. Then the compressed DCT components are rotated, otherwise ICA cannot be applied because all the DCT components have energy in the lower frequencies and they are highly correlated to each other. The rotations have another aspect. The directions and degrees of the rotations are saved as “key” to restore the original images. If the receiver does not know “key,” it is hard to restore the original images.

In order to validate our approach several computer simulations are conducted. Varying the sizes of DCT blocks and the selected information, the performance of the compression are evaluated using root mean square error (RMSE) between the original images and the reconstructed images. The best RMSEs are obtained when the compression ratio is between 0.1 and 0.5.

II. Independent Component Analysis

In the last decade, several independent component analysis (ICA) algorithms have been proposed. ICA is a method to find underlying independent components from statistical data. ICA is often used to solve blind source separation (BSS) problem, which means retrieving unknown sources (signals or images) by only observing mixtures of them. In general, it is assumed that the sources are non-Gaussian and statistically independent of each other,

Here we formulate the BSS problem. We have M unknown sources and N observed mixed signals. It is generally supposed that N is greater than or equal to M, however,

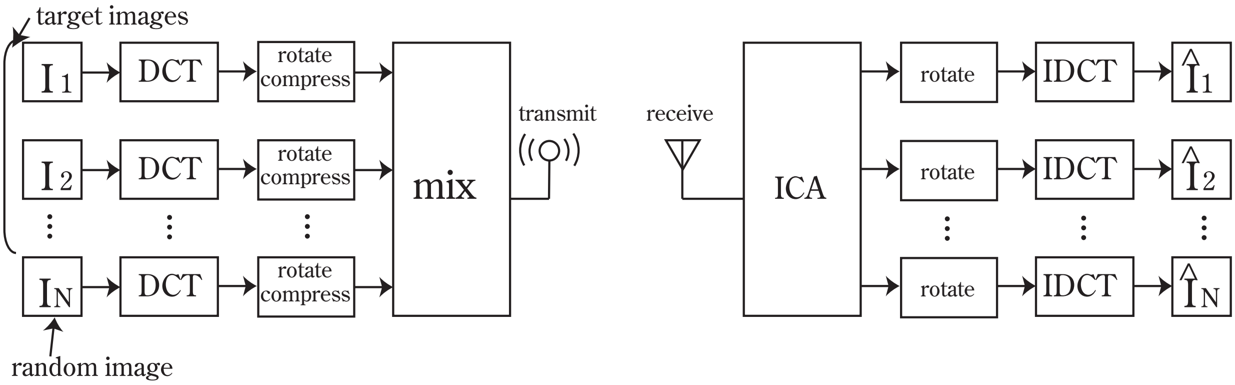


Fig. 1. Our approach for encryption and compression

complete mixture case ($M = N$) is considered in this paper. We define sources as $\boldsymbol{s}(n) = [s_1(n), \ldots, s_N(n)]^T$, where ^{T} denotes a transpose of a vector or matrix. Then mixed signals $\boldsymbol{x}(n) = [x_1(n), \ldots, x_N(n)]^T$ can be denoted as

$$\boldsymbol{x}(n) = \boldsymbol{H}\boldsymbol{s}(n), \tag{1}$$

where $\boldsymbol{H}$ is an $N \times N$ mixing matrix whose element h_{ij} is a real or complex coefficient.

To achieve blind source separation, we must estimate a separating matrix $\boldsymbol{W}$, without knowledge about sources and their mixing process, such that the output signals $\boldsymbol{y}(n) = [y_1(n), \ldots, y_N(n)]$ should be independent of each other,

$$\boldsymbol{y}(n) = \boldsymbol{W}\boldsymbol{x}(n). \tag{2}$$

To estimate the separating matrix $\boldsymbol{W}$, we can find various ICA algorithms [8]. These algorithms generally use different approaches:

- the minimization of a cost function based on the Higher-Order Statistics (HOS) [9], [10]
- the maximization of mutual information [11]
- using geometrical concepts [12]
- using nonstationarity of sources [13], etc.

In this paper we use FastICA algorithm [10], which is one of the most popular ICA algorithms, to estimate a separating matrix.

Using ICA algorithms we can estimate independent sources up to permutation and scaling ambiguities, that is,

$$\boldsymbol{y}(n) = \boldsymbol{D}\boldsymbol{P}\boldsymbol{s}(n), \tag{3}$$

where $\boldsymbol{D}$ is a diagonal matrix and $\boldsymbol{P}$ is a permutation matrix. $\boldsymbol{D}$ and $\boldsymbol{P}$ mean scaling and permutation indeterminacy, respectively.

III. Approach

Flow of our approach to transmit encrypted and compressed images to authorized people is illustrated in Fig. 1. In the encryption stage, we divide the images into small blocks and apply two-dimensional Discrete Cosine Transform (DCT) separately of each block. With the property of the DCT transform which regroup the important information in the upper-left corner, we can easily select this desired information by applying a simple low pass filter to compress them. Thus we can reduce the size of information to be transmitted.

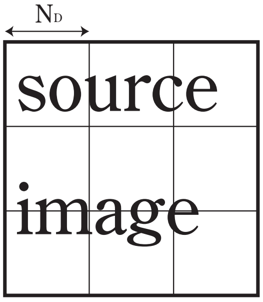


Fig. 2. dividing original image into small blocks

After the selection of desired information, we will encrypt them by grouping together (merging) the information coming form several DCT components of the images in the spectral plane. As we will use ICA at receiver in order to retrieve the different DCT components, we must pay attention to the dependence of these DCT components. Because the DCT components of natural images have high energy in lower frequency bands, DCT components of different sources may not be independent. To resolve this problem and obtain an independent version of these DCT components, we rotate each DCT block randomly. Moreover, the rotation operation will be used as an adding encryption key. This latent key must be sent to the receiver in order to rebuild the decrypted images.

In the decryption stage, the receiver applies ICA to the mixed image, then rotates the DCT components based on information of the encryption keys. Finally applying Inverse Discrete Cosine Transform (IDCT) to the reconstructed DCT components, the original images can be reconstructed.

In the following subsections, we explain the detail of the processes.

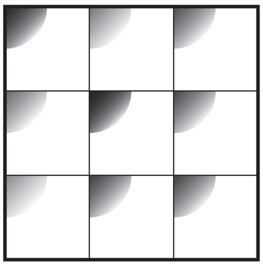


Fig. 3. DCT components of an image

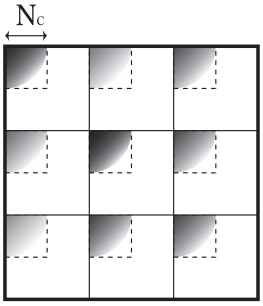


Fig. 4. Compression of DCT components

A. Encryption

1) Discrete Cosine Transform: At first we divide original images to be transmitted into small square blocks (Fig. 2) and apply two-dimensional discrete cosine transform to each block and we obtain DCT components of each block as shown in Fig. 3. In the following simulations, the size of DCT blocks, N_D is varied and the performances are evaluated.

2) Compression: For a fast communication, we would like to reduce the amount of transmitting data. Consequently, compression of the DCT components is required. In each block, most of DCT components have high energies in low frequency bands, we only use low frequency components through a simple low pass filter (Fig. 4), that is, left-up corners of each block with size of $N_C \times N_C$ are selected and higher frequency components are dropped.

As a result of this process we can compress the transmitting images. However, loss of high frequency components may cause degrading the image quality. Effect of the compression size is also investigated in the simulations.

3) Rotation: After the compression we have to rotate the small blocks randomly, because our approach is based on independent component analysis. To apply ICA algorithms, sources must be statistically independent of one another. However compressed DCT components of natural images are not independent since every small DCT block has a high energy in lower frequency bands. Therefore, in order to overcome this problem we rotate each block randomly, so as to make rotated DCT components be independent of each other. DCT blocks are rotated in 8 different patterns including no rotation (see Fig. 5). The rotations are chosen so as not to change the size of DCT blocks.

1	2	3
4	5	6
7	8	9

(a) No rotation (original)

1	4	7
2	5	8
3	6	9

(b) Rotation 1

7	4	1
8	5	2
9	6	3

(c) Rotation 2

3	2	1
6	5	4
9	8	7

(d) Rotation 3

9	8	7
6	5	4
3	2	1

(e) Rotation 4

9	6	3
8	5	2
7	4	1

(f) Rotation 5

3	6	9
2	5	8
1	4	7

(g) Rotation 6

7	8	9
4	5	6
1	2	3

(h) Rotation 7

Fig. 5. Rotation patterns

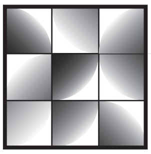


Fig. 6. Compressed and Rotated DCT components

By means of the rotation we can obtain the rotated DCT components of source images, which are statistically independent of each other (see Fig. 6). Thus we can apply ICA algorithm to their mixtures.

When we rotate each block, we save the rotation pattern of each block. The rotation data will be used as "key" to restore the image. Without the "key" it might be hard to reconstruct the original image from the rotated DCT components.

4) Mixing: Our encryption is inspired by Blind Source Separation. Compressed and rotated DCT blocks are mixed together. For example, to encrypt the image of "Lena" we utilize a random image. When we mix Lena and a random

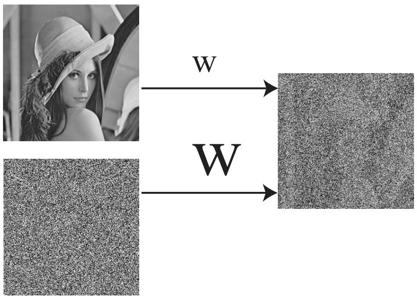


Fig. 7. Mixing images

image, we put higher weight on the random image so that "Lena" image should be hidden in the mixtures (see Fig. 7). Even if non-authorized people intercept the mixed images, the mixtures seem the random images. In the actual process, original images themselves are not mixed but their DCT components are used.

In this stage, after the compression and the rotation of multiple images we mix them and transmit the mixture. The receiver observes the mixed DCT components.

B. Decryption

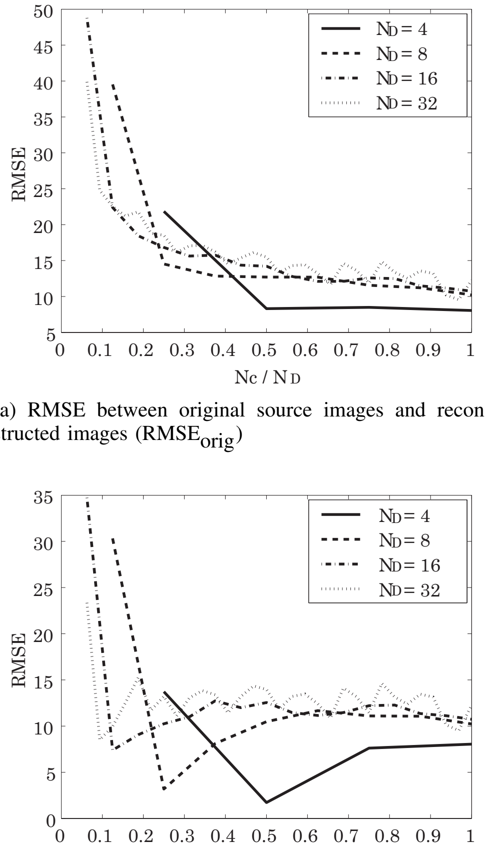
Authorized people receive the mixtures and demix them. Turning back the rotated DCT components and using inverse discrete cosine transform (IDCT), original images are decrypted.

1) Demixing: Authorized people receive the mixtures of the DCT components of the original images. At first the receiver has to demix the mixtures of the DCT components, i.e. solve the BSS problem. In order to achieve BSS, Independent Component Analysis (ICA) algorithm is applied. In this paper FastICA [10] is used because it is so popular due to its property of fast convergence. Thus the demixed images, which are composed of compressed and rotated DCT components, can be obtained.

2) Inverse rotation and inverse discrete cosine transform: After separation of the mixed DCT components, original images are reconstructed. The rotated DCT components have to be restored. The receiver is beforehand given the rotation "key" by the transmitter. Based on the rotation "key", the receiver can reconstruct the original images, rotating the DCT components contrary to the encryption stage. Without rotation key, it is difficult to reconstruct the original DCT components. Finally he can apply inverse discrete cosine transform (IDCT) to them. In this way the receiver can obtain the estimated original images from the observed mixtures.

IV. SIMULATION

In order to validate our approach several simulations are conducted. "Lena" and "Mandrill" images are encrypted and decrypted by the proposed method. 256×256 grayscale bitmap files are used as the original images. An example of the simulations, mixing three images including a random image, is shown in Fig. 8. The first row of Fig. 8 shows the original source images ((a)–(c)), the images in the second row were obtained by applying IDCT to the mixed DCT components ((d)–(f)), and the last row corresponds the reconstructed images which the receiver can obtain ((g)–(i)). As shown in the 2nd row even if non-authorized people apply IDCT to the mixed images, obtained images seem insignificant. After execution of ICA and applying IDCT to the separated DCT components with the rotation key, we can get the source images. The quality of the reconstructed images, however, is not as same as the original ones, because compression cut off higher frequency components and ICA cannot separate the images completely. Moreover, the order of the outputs is not always same as that of the original images owing to permutation ambiguity of ICA.



(a) RMSE between original source images and reconstructed images ($\mathrm{RMSE_{orig}}$)

(b) RMSE between compressed source images and reconstructed images ($\mathrm{RMSE_{comp}}$)

Fig. 9. RMSEs for two images

Here, in order to evaluate the performance of the proposed method. we quantized the performance using root mean square error (RMSE), which is defined as

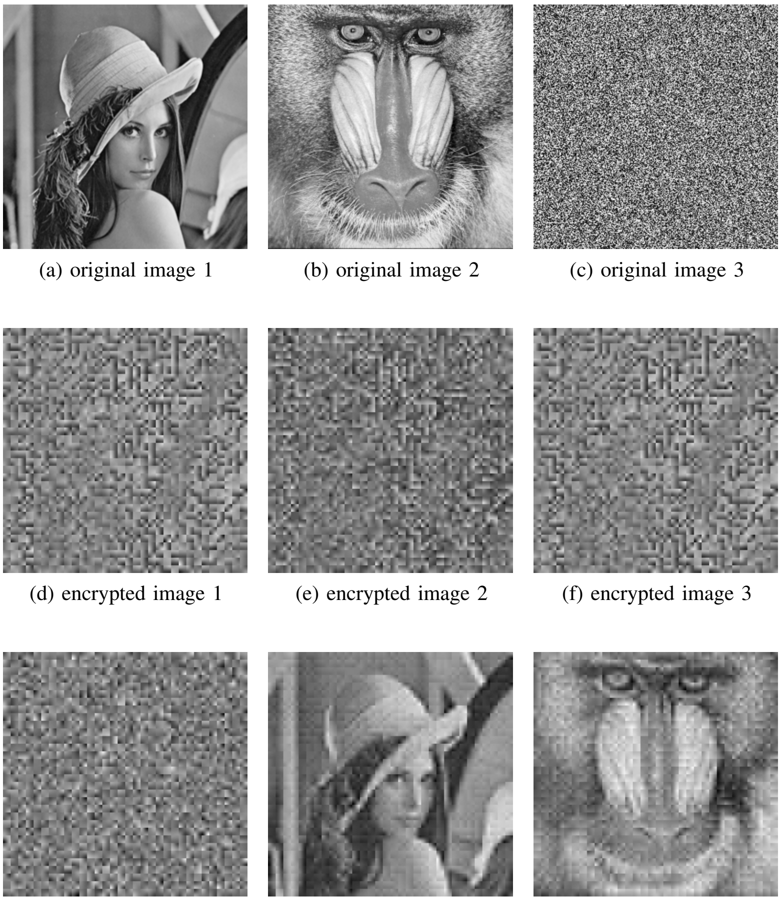
$$\mathrm{RMSE_{orig}} = \sqrt{\frac{1}{L^2}\sum_{i,j=1}^{L}\Big(\mathrm{I_{orig}}(i,j) - \alpha\mathrm{I_{est}}(i,j)\Big)^2}, \quad (4)$$

where $\mathrm{I_{orig}}(i,j)$ is (i,j)th pixel of the original image to be transmitted, $\mathrm{I_{est}}(i,j)$ is that of the restored image, L is a size of images, i.e. 256, and α is the scaling factor which is caused by ICA, because the outputs of ICA contains the scaling indeterminacy as mentioned before.

Using the proposed method the higher frequency components are cut off, that is, the quality of the original image is reduced. Therefore $\mathrm{RMSE_{comp}}$ is also defined

$$\mathrm{RMSE_{comp}} = \sqrt{\frac{1}{L^2}\sum_{i,j=1}^{L}\big(\mathrm{I_{comp}}(i,j) - \alpha\mathrm{I_{est}}(i,j)\big)^2}, \quad (5)$$

where $\mathrm{I_{comp}}(i,j)$ is also the (i,j)th pixel of the compressed original image, which is obtained by applying IDCT to the compressed DCT components. $\mathrm{RMSE_{comp}}$ indicates the



(a) original image 1 (b) original image 2 (c) original image 3

(d) encrypted image 1 (e) encrypted image 2 (f) encrypted image 3

(g) reconstructed image 1 (h) reconstructed image 2 (i) reconstructed image 3

Fig. 8. Example of the simulation; 1st row: original images; 2nd row: IDCT of mixed DCT components; 3rd row: Restored images

performance excludes compression process, while $\text{RMSE}_{\text{orig}}$ shows the performance of the whole process.

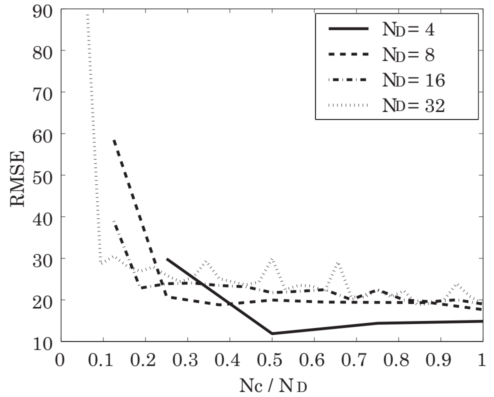
Varying DCT size N_D and compression (smaller) block size N_C, we calculate RMSEs. Using two images, Lena image and a random image, the results are shown in Fig. 9 and using three images, Lena, Mandrill and a random images, in Fig. 10. In the figures horizontal axis denotes the ratio of compression size to DCT size, vertical axis is RMSE. Solid lines show the results in the cases where N_D is 4, dashed lines 8, dotted-dashed lines 16, and dotted lines 32. The smaller DCT block size we use, the smaller RMSE is. When the ratio of compression size to DCT size is from 0.1 to 0.5, the best RMSEs were obtained. If the compression size is too large, most of DCT components are near zero and independent assumption is no longer satisfied. Conversely if the compression is small, only the low frequency components are left and they also have high correlations. Therefore we have to choose an optimal compression size.

V. Conclusion

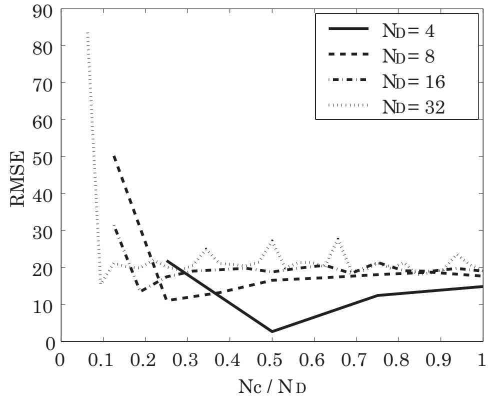
In this paper we proposed a new image encryption and compression method based on Independent Component Analysis (ICA) and Discrete Cosine Transform (DCT). Using DCT and a simple low pass filter, images can be compressed. For encryption, DCT blocks of transmitted images are rotated and mixed with a random image to hide them. In the decryption stage, the covered images can be extracted from the mixtures by applying ICA. Finally using rotation keys and inverse discrete cosine transform, the original images can be reconstructed. Therefore we can achieve a fast and secure image transmission.

As a result of several computer simulations, the behavior of the proposed approach is confirmed. When compression ratio is between 0.1 and 0.5, the best performance can be achieved. In this paper only grayscale images are used as original images, but color images can be applied in the same way.

Our future works include a more secure encryption method with an alternative rotation method and a reconstruction key. More complex rotation manner makes it harder for non-authorized people to reconstruct images without keys. In this paper only 8 patterns are used. Our next challenge is transmitting a video sequence, which is composed of a series of similar images. In such a case, the problem about statistical independence of source images occurs to apply some



(a) RMSE between original source images and reconstructed images ($\text{RMSE}_{\text{orig}}$)



(b) RMSE between compressed source images and reconstructed images ($\text{RMSE}_{\text{comp}}$)

Fig. 10. RMSEs for three images

independent component analysis algorithm. We expect that using the proposed method, this problem is resolved.

References

[1] A. Hyvärinen, J. Karhunen, and E. Oja, *Independent Component Analysis*. John Wiley & Sons, Inc., 2001.

[2] A. Cichocki and S. Amari, *Adaptive Blind Signal and Image Processing*. John Wiley & Sons, Inc., 2002.

[3] Y. Deville, "Towards industrial applications of blind soure separation and independent component analysis," in *Proc. ICA99*, 1999, pp. 19–24.

[4] A. Mansour, M. Kawamoto, and N. Ohnishi, "A survey of the performance indices of ICA alrotihms," in *Proc. 21st IASTED International Conference on Modeling, Identification and Control*, 2002, pp. 660–666.

[5] E. Chaumette, P. Comon, and D. Muller, "Application of ICA to airport surveillance," in *Proc. HOS93*, 1993, pp. 210–214.

[6] I. Kopriva and A. Persin, "Blind separation of optical tracker responses into independent components discriminates optical sources," in *Proc. ICA99*, 1999, pp. 31–36.

[7] A. Alfalou and A. Mansour, "All optical video-image encryption enforced security level using ica," *Journal of Optics A: Pure and Applied Optics*, vol. 9, pp. 787–796, 2007.

[8] A. Mansour, A. K. Barros, and N. Ohnishi, "Blind separation of sources: Methods, assumptions and applications," *IEICE Trans. Fundamentals*, vol. E83-A, no. 8, pp. 1498–1512, 2000.

[9] A. Mansour and N. Ohnishi, "Multichannel blind separation of sources algorithm based on cross-cumulant and the levenberg-marquardt method," *IEEE Trans. Signal Process.*, vol. 47, no. 11, pp. 3172–3175, 1999.

[10] A. Hyvärinen, "Fast and robust fixed-point algorithms for independent component analysis," *IEEE Trans. on Neural Networks*, vol. 10, no. 3, pp. 626–634, 1999.

[11] A. J. Bell and T. J. Sejnowski, "An information-maximization approach to blind separation and blind deconvolution," *Neural Computation*, vol. 7, no. 6, pp. 1129–1159, 1995.

[12] A. Mansour, N. Ohnishi, and C. G. Puntonet, "Blind multiuser separation of instantaneous mixture algorithm based on geometrical concepts," *Signal Processing*, vol. 82, no. 8, pp. 1155–1175, 2002.

[13] K. Matsuoka, M. Ohya, and M. Kawamoto, "A neural net for blind separation of nonstationary signals," *Neural Networks*, vol. 8, no. 3, pp. 411–419, 1995.