신경회로망 ICA를 이용한 혼합영상신호의 분리

論 文
57-8-23

Blind Image Separation with Neural Learning Based on Information Theory and Higher-order Statistics

曺 賢 哲* · 李 權 純†
(Hyun Cheol Cho · Kwon Soon Lee)

Abstract – Blind source separation by independent component analysis (ICA) has applied in signal processing, telecommunication, and image processing to recover unknown original source signals from mutually independent observation signals. Neural networks are learned to estimate the original signals by unsupervised learning algorithm. Because the outputs of the neural networks which yield original source signals are mutually independent, then mutual information is zero. This is equivalent to minimizing the Kullback-Leibler convergence between probability density function and the corresponding factorial distribution of the output in neural networks. In this paper, we present a learning algorithm using information theory and higher order statistics to solve problem of blind source separation. For computer simulation two deterministic signals and a Gaussian noise are used as original source signals. We also test the proposed algorithm by applying it to several discrete images.

Key Words : Blind image separation, ICA, Neural network, Information theory, Higher-order statistics

1. Introduction

Independent component analysis (ICA) is a statistical approach to express a set of random variables or signals in terms of statistically independent components. The main purpose of ICA is to recover independent original sources or signals from observation signals by sensor systems. Popular applications of ICA are blind source (signal) separation, blind deconvolution, feature extraction, and noise cleaning. Blind source separation can be applied to many areas such as in signal processing, data communication, speech recognition, and medical science [1][9]. In a large number of applications, signals received by sensors such as antennas or microphone systems are a mixture of original source signals. Generally, these source signals are usually unknown in case of acoustic signal processing, radar array or sonar signal processing, and biomedical signal processing [1][2][4][5][6][7][9]. By using only the mixed signals observed by sensor systems, original source signals are estimated or separated. Therefore, blind source separation may be represented as estimating or separating original sources without knowing the characteristics of the transmission systems. Fig. 1 shows a block diagram for blind source separation. An unknown signal vector s with statistically independent components is defined by

$$s = [s_1, s_2, \cdots, s_m]^T \tag{1}$$

A unknown observation signal vector $x \in R^m$ is expressed by

$$x = As \tag{2}$$

where $A \in R^{m \times m}$ is a unknown nonsingular mixing matrix. Both source vector s and mixing matrix A are unknown, but an observation vector x is known. Multiplying a demixing matrix W by an observation vector x, we have the separated signals defined by

$$y = Wx \in R^m \tag{3}$$

where $W \in R^{m \times m}$ is modeled by neural networks as shown in Figure 2. The inputs of neural networks as a demixer are observation signals and the outputs are the separated signals. To estimate original signals, neurons in the networks are learned by unsupervised learning algorithm [10][12]. Since unknown original sources are zero mean and statistically independent, therefore, both observation signals x and output signals y are zero mean and statistically independent signals likewise. Several

* 正 會 員 : 동아대학교 전기공학과 포닥연구원
† 교신저자, 正會員 : 동아대학교 전기공학과 교수 · 공박
E-mail : kslee@dau.ac.kr
接受日字 : 2007年 11月 22日
最終完了 : 2008年 4月 9日

approaches based on higher order statistics [13] or information theory such like negentropy [14], maximum entropy [15], mutual information [16], and infomax [17] as well as maximum likelihood estimation [18], recently have been developed for blind source separation. In this paper, we describe neural learning algorithm using mutual information theory and higher order moments to separate blind image signals.

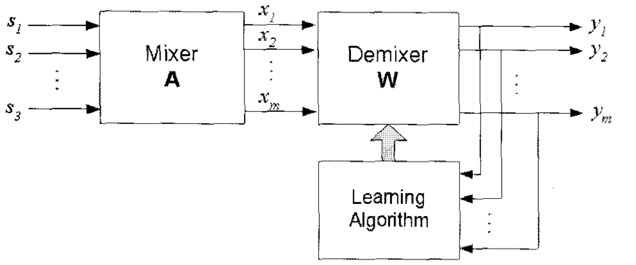


Fig. 1 Processing of blind source separation

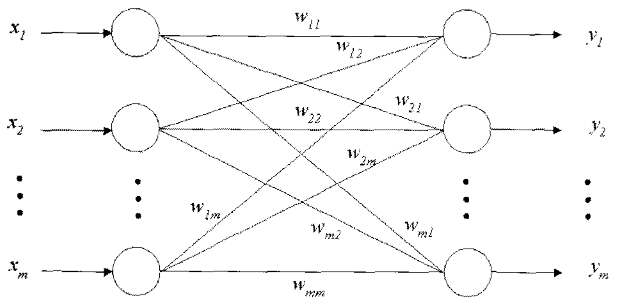


Fig. 2 Neural network for blind source separation

2. Mutual Information

The amount of information obtained after the occurrence of x with probability p in a random variable X is defined by [19]

$$I(x) = \log\left(\frac{1}{P(x)}\right) = -\log p(x) \tag{4}$$

The entropy of X means value of the amount of information associated with X. The entropy of a continuous random variable with probability density function $f_X(x)$ is given by

$$h(X) = E[I(x)] = -\int_{-\infty}^{\infty} f_x(x) \log f_x(x)\, dx \tag{5}$$

where E denotes the expectation. The entropy for a random vector X consisting m random variables is defined by

$$h(X) = -\int_{-\infty}^{\infty} f_x(x) \log f_x(x)\, dx \tag{6}$$

We define the conditional entropy of a random vector X given Y as

$$h(X|Y) = h(X,\ Y) - h(Y) \tag{7}$$

where $h(X,Y)$ is the joint entropy of random vectors X and Y

$$h(X,Y) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f_{x,y}(x,y) \log f_{x,y}(x,y)\, dx dy \tag{8}$$

and $f_{X,Y}(x,y)$ is the joint probability density function of X and Y. The entropy $h(X)$ is a measure of uncertainty about a system input before observing the system output, and the conditional entropy $h(X|Y)$ is a measure of uncertainty about a system input after observing the system output. We define the difference between $h(X)$ and $h(X|Y)$ as the mutual information $I(X;Y)$ between X and Y

$$\begin{aligned} I(X;Y) &= h(X) - h(X|Y) \\ &= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f_{x,y}(x,y) \log\left(\frac{f_x(x|y)}{f_x(x)}\right) dx dy \end{aligned} \tag{9}$$

By the definition of conditional probability we have

$$f_x(x|y) = \frac{f_{X,Y}(x,y)}{f_Y(y)} \tag{10}$$

then rewrite (11) to

$$\begin{aligned} I(X;Y) &= h(X) - h(X|Y) \\ &= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f_{x,y}(x,y) \log\left(\frac{f_x(x|y)}{f_x(x) f_y(y)}\right) dx dy \end{aligned} \tag{11}$$

The mutual information between X and Y is equal to the Kullback-Leibler divergence between the joint probability density function $f_{X,Y}(x,y)$ and the product of the probability density functions $f_X(x)$ and $f_Y(y)$. We use the special case that is the Kullback-Leibler divergence between the probability density function $f_X(x)$ and the product of its marginal probability density functions $f_{X_i}(x_i)$ [20]

$$D_{f_X \| f_{X_i}} = \int_{-\infty}^{\infty} f_X(x) \log\left(\frac{f_X(x)}{\Pi_{i=1}^{m} f_{x_i}(x_i)}\right) dx \tag{12}$$

The ith marginal probability density function is defined by

$$f_{x_i}(x_i) = \int_{-\infty}^{\infty} f_X(x) dx^{(i)} \tag{13}$$

where $dx^{(i)}$ is a $(m-1)\times 1$ vector left after removing the ith element of a vector x. We rewrite (14) in the expanded form

$$D_{f_X \| f_{X_i}} = \int_{-\infty}^{\infty} f_X(x) \log f_X(x) dx - \sum_{i=1}^{m} \int_{-\infty}^{\infty} f_X(x) \log_{X_i}(x_i) dx \tag{14}$$

Because of $dx = dx^{(i)} dx_i$, we rewrite in the second integral term of (12)

$$\int_{-\infty}^{\infty} f_X(x) \log f_X(x_i) dx = \int_{-\infty}^{\infty} \log f_{X_i}(x_i) dx \int_{-\infty}^{\infty} f_X(x) dx^{(i)} dx_i \tag{15}$$

By using (13), we rewrite (15) as

$$\int_{-\infty}^{\infty} f_X(x) \log f_{X_i}(x_i) dx = \int_{-\infty}^{\infty} f_{X_i}(x_i) \log f_{X_i}(x_i) dx_i = -h_{X_i}(x_i) \tag{16}$$

Finally, we simplify the Kullback-Leibler divergence in (14) by substituting (6) and (16) to obtain the objective (contrast) function for blind source separation [21]

$$D_{f_X \| f_{X_i}} = -h(X) + \sum_{i=1}^{m} h_{x_i}(x_i) \tag{17}$$

where $h_{X_i}(x_i)$ is the ith marginal entropy based on the marginal probability density function.

3. Learning of Neural Networks

We rewrite the Kullback-Leibler divergence by applying (3) to (17)

$$D(W) = -h(y) + \sum_{i=1}^{m} h_{Y_i}(y_i) \tag{18}$$

Using properties of entropy (see Appendix I), the entropy $h(y)$ in (18) is expressed by

$$h(y) = h(x) + \log|\det(W)| \tag{19}$$

where $\det(W)$ is the determinant of W. To calculate $h_{Y_i}(y_i)$ we need the marginal distribution of y_i and must then integrate out the effects of all the components of the random vector y. But it is usually difficult to calculate $h_{Y_i}(y_i)$ because of high dimensionality of y. We derive an approximate expression for the marginal probability density function $f_{Y_i}(y_i)$ in terms of higher-order moments of y using the Gram-Charlier expansion [20].

$$f_{Y_i}(Y_i) \approx \alpha(y_i) \left\{ 1 + \frac{K_{i,3}}{3!} H_3(y_i) + \frac{K_{i,4}}{4!} H_4(y_i) \right\} \tag{20}$$

where $\alpha(y_i)$ is the probability density function of a normalized Gaussian random variable with zero mean and unit variance, i.e.

$$\alpha(y_i) = \frac{1}{\sqrt{2\pi}} exp\left(\frac{y_i^2}{2} \right) \tag{21}$$

and $H_k(y_i)$ are Chebyshev Hermite polynomials. The kth order cumulant of y_i [21] is given by

$$K_{i,1} = E(y_i) \tag{22-1}$$

$$K_{i,2} = E(y_i^2) - (E(y_i))^2 \tag{22-2}$$

$$K_{i,3} = E(y_i^3) - 3E(y_i)E(y_i^2) + 2(E(y_i))^3 \tag{22-3}$$

$$K_{i,4} = E(y_i^4) - 3(E(y_i^2))^2 - 4E(y_i^3)E(y_i) + 12E(y_i^2)(E(y_i))^2 - 6(E(y_i))^4 \tag{22-4}$$

Finally by applying the expansion of a logarithm we have

$$\log(1+c) \approx c - \frac{c^2}{2} + O(c^3) \tag{23}$$

and using properties of the Chebyshev Hermite polynomials, we may expand the marginal entropy in the right hand side of (18) [15]

$$h_{yi}(y_i) \approx \frac{1}{2} log(2\pi e) - \frac{(k_{i,3})^2}{12} - \frac{(k_{i,4})^2}{48} + \frac{5}{8}(k_{i,3})^2 k_{i,4} + \frac{1}{16}(k_{i,4})^3 \tag{24}$$

By substituting (19) and (24) to (18), we have

$$D(W) \approx -h(x) - \log|\det(W)| + \frac{m}{2}\log(2\pi e) - \sum_{i=1}^{m} \left(\frac{(k_{i,3})^2}{12} + \frac{(k_{i,4})^2}{48} + \frac{5(k_{i,3})^2 k_{i,4}}{8} + \frac{(k_{i,4})^3}{16} \right) \tag{25}$$

We differentiate the objective function in (25) with respect to W to derive a neural network learning algorithm. Assuming zero mean, derivatives of terms in (25) are

$$\frac{\partial}{\partial w_{ik}} log|\det(W)| = (W^{-T})_{ik} \tag{26}$$

$$\frac{\partial}{\partial w_{ik}}(k_{i,3})^2 = 6E(y_i^3)E(y_i^2)x_k \tag{27}$$

$$\frac{\partial}{\partial w_{ik}}(k_{i,4})^2 = 8E(y_i^4)E(y_i^3)x_k - 24E(y_i^3)x_k \tag{28}$$

$$\frac{\partial}{\partial w_{ik}}(k_{i,3})^2 k_{i,4} = 6E(y_i^4)E(y_i^3)E(y_i^2)x_k - 18E(y_i^3)E(y_i^2)x_k + 4(E(y_i^3))^3 x_k \tag{29}$$

$$\frac{\partial}{\partial w_{ik}}(k_{i,4})^3 = 12(E(y_i^4)-3)^2 E(y_i^3)x_k \tag{30}$$

where w_{ik} denotes the ikth cofactor in W(see Appendix II for proofs of (27)-(30)). Replacing expectations with their instantaneous values we obtain

$$\frac{\partial}{\partial w_{ik}}(k_{i,3})^2 = 6y_i^5 x_k \tag{31}$$

$$\frac{\partial}{\partial w_{ik}}(k_{i,4})^2 = (8y_i^7 - 24y_i^3)x_k \tag{32}$$

$$\frac{\partial}{\partial w_{ik}}(k_{i,3})^2 k_{i,4} = (10y_i^9 - 18y_i^5)x_k \tag{33}$$

$$\frac{\partial}{\partial w_{ik}}(k_{i,4})^3 = (12y_i^{11} - 72y_i^7 + 108y_i^3)x_k \tag{34}$$

Using (31)-(34), we obtain the derivative of (25) is

$$\frac{\partial}{\partial w_{ik}} D(W) \approx -(W^{-T})_{ik} + \Phi(y_i)x_k \tag{35}$$

where $\phi(y_i)$ is an activation function for neural networks learning given by

$$\phi(y_i) = \frac{29}{4}y_i^3 - \frac{47}{4}y_i^5 - \frac{14}{3}y_i^7 + \frac{25}{4}y_i^9 + \frac{3}{4}y_i^{11} \tag{36}$$

The objective of the learning algorithm is to minimize the Kullback-Leibler divergence in (25), and is implemented using a gradient descent adjustment of the weights of neural networks

$$\begin{aligned}\Delta w_{ik} &= -\eta \frac{\partial}{\partial w_{ik}} D(W) \\ &= \eta((W^{-T})_{ik} - \phi(y_i)x_k)\end{aligned} \tag{37}$$

where η is learning rate. We rewrite (37) in a matrix form

$$\Delta W = \eta(W^{-T} - \phi(y)x^T) \tag{38}$$

where $\phi(y)$ is a column vector of an activation function

$$\phi(y) = [\Phi(y_1), \Phi(y_1), \cdots, \Phi(y_m)]^T \tag{39}$$

By applying $y^T = x^T W^T$ to (40), we obtain

$$\begin{aligned}\Delta W &= \eta[I - \Phi(y)x^T W^T] W^{-T} \\ &= \eta[I - \Phi(y)x^T] W^{-T}\end{aligned} \tag{40}$$

Because a mixing matrix A is nonsingular, it is more useful to describe (40) by a natural gradient descent algorithm [22] as

$$\Delta W = -\eta \frac{\partial D(W)}{\partial W} W^T W \tag{41}$$

We finally write the update rule for adapting neural network weighs

$$\begin{aligned}W(k+1) &= W(k) + \eta[I - \Phi(y(k))y^T](W(k)W^T(k))W^{-T}(k) \\ &= W(k) + \eta[I - \Phi(y(k))y^T(k)]W(k)\end{aligned} \tag{42}$$

4. Computer Simulation

We use two images as original sources shown in Fig. 3 and its histograms are provided in Fig. 4. We normalize the gray value for each pixel. Fig. 5 shows the waveform of the normalized signals (1st and 2nd). The mixing matrix A in this simulation is selected by

$$A = \begin{bmatrix} 0.2 & 0.5 \\ 0.4 & 0.3 \end{bmatrix} \tag{43}$$

The image mixed with the mixing matrix (43) is shown in Fig. 6 and its waveforms are shown in Fig. 5 (3rd and 4th). We select the initial matrix

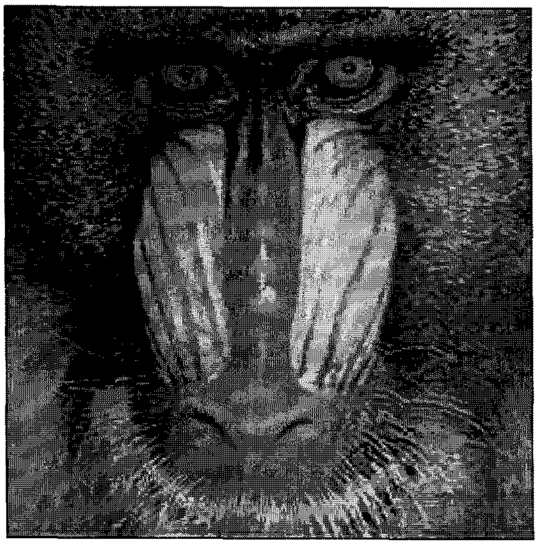
$$W = \begin{bmatrix} 0.7294 & -0.3546 \\ -0.2323 & 0.4726 \end{bmatrix} \tag{44}$$

After 393216 learning iterations, we obtained the best result and the corresponding separated signals shown in Fig. 7. The 5th and 6th waveforms in Fig. 5 show the normalized waveforms of the separated images. Next, we use only one image mixed with a Gaussian random noise. Fig. 8 shows the mixed images applying the mixing matrix of (43). The initial demixing matrix is also the same as (44). The separated signals are shown in Fig. 9. This application is used to a noise extraction. Lastly, we have two original images in Fig. 10 and the histograms of the original images are shown in Fig. 11. The mixing matrix is similar to (43) and the initial weight matrix is

$$W=\begin{bmatrix} 0.1182 & -0.0321 \\ -0.0398 & 0.0513 \end{bmatrix} \tag{45}$$

Figs. 12 and 13 show the mixing images and the separated images, respectively. As seen, the separated image of the left side in Fig. 13 is the inversion of the original image. We can easily restore the image using simple inversion. The restored image is shown in Fig. 14. We provide a qualitative comparison to a well known blind source separation approach [23]. The authors in [23] addressed the blind separation algorithm with faster convergence property, which has quadratic or cubic convergence, whereas the gradient method in our proposed algorithm has only linear convergence. However, the result in [23] is somewhat theoretical since the learning is likely to be local convergence. Thus, the latter is hardly implemented for online separation algorithm, but our algorithm is computational efficiency and applicable in practice.

(a)



(b)

Fig. 3 Original images

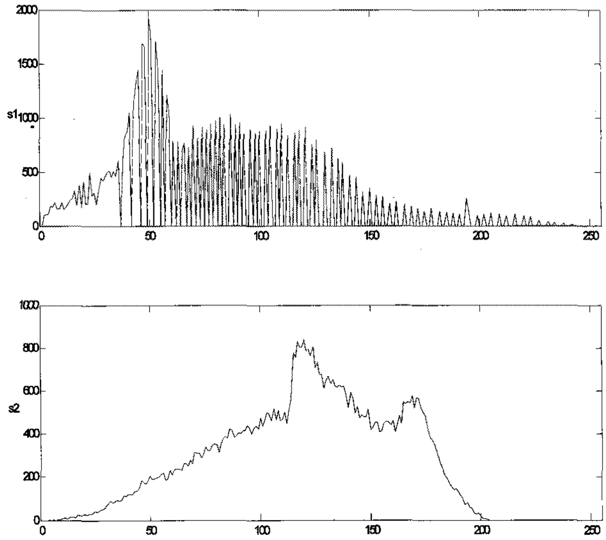


Fig. 4 Histogram of original images

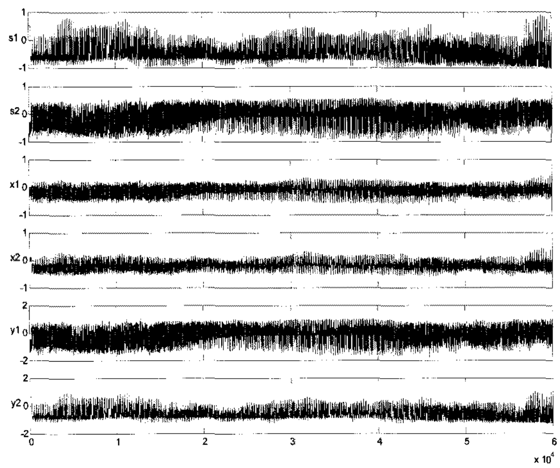
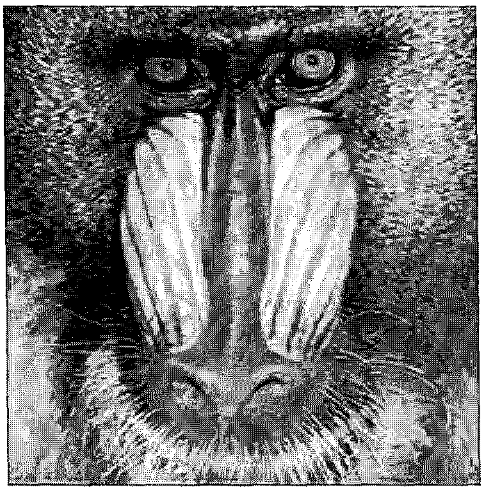


Fig. 5 Waveform of normalized source signals

(a)

(b)

Fig. 6 Mixing images

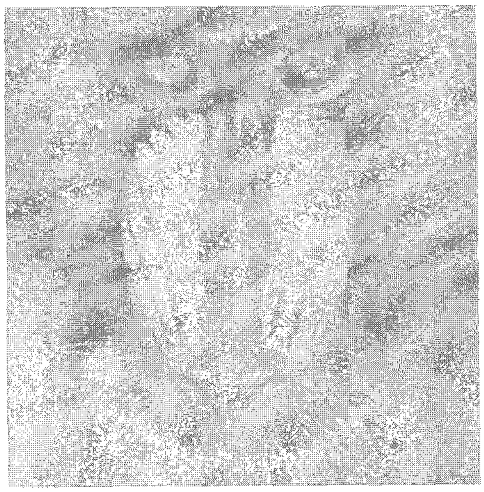


(a)

(b)

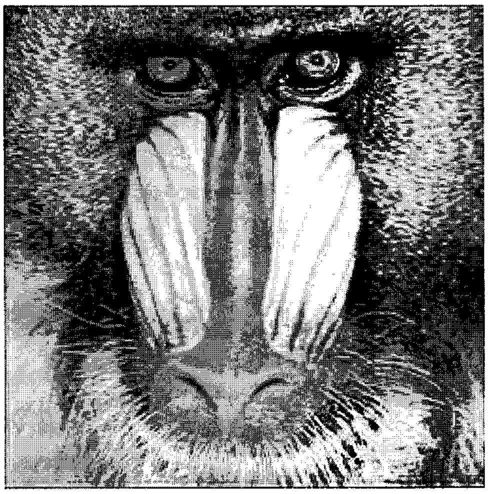
Fig. 7 Separated images

(a)

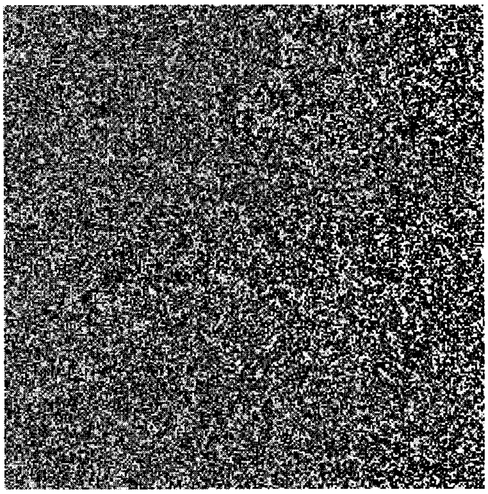


(b)

Fig. 8 Mixing images with a Gaussian noise



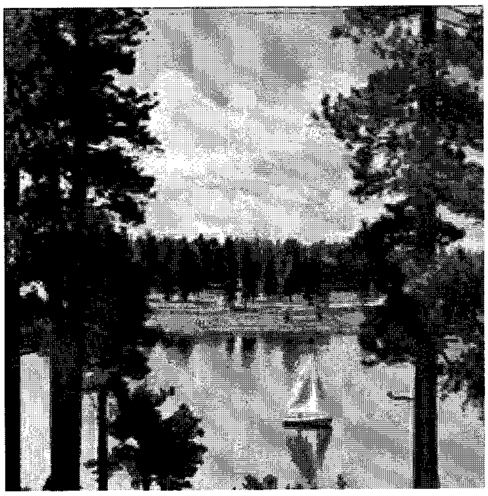
(a)



(b)

Fig. 9 Separated original image and noise signal

(a)



(b)

Fig. 10 Original images

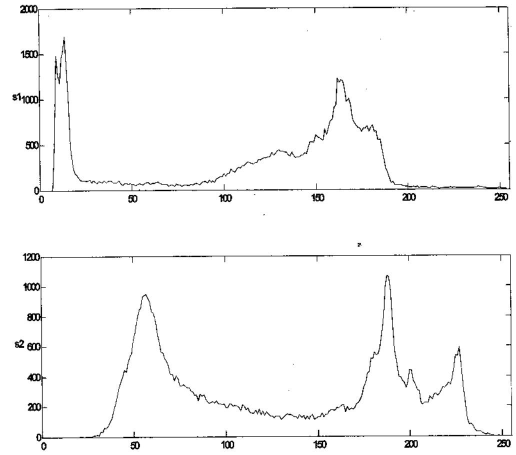
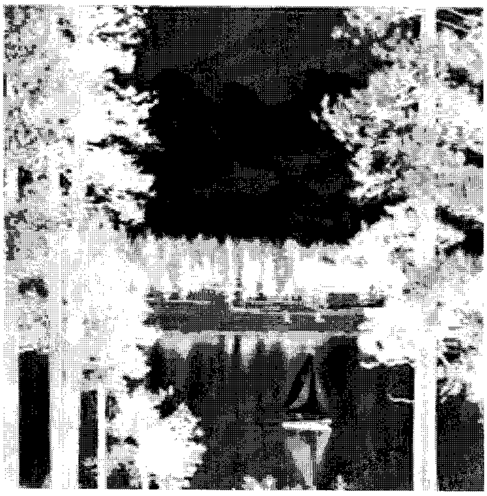


Fig. 11 Histogram of original images

(a)

(b)

Fig. 12 Mixing images



(a)

(b)

Fig. 13 Separated images

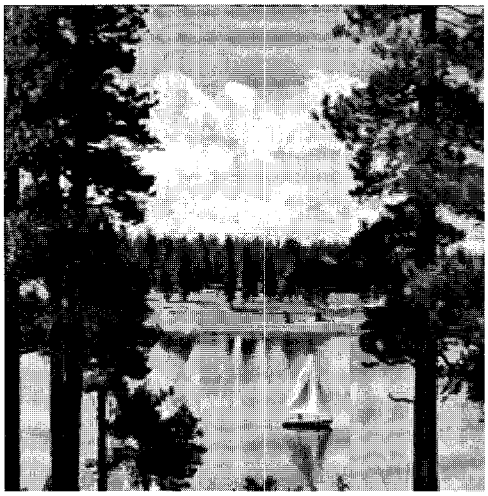


Fig. 14 Inversion of separated image

5. Conclusions

We propose an approach to blind source separation using information theory and higher order statistics. Because the separated signals are assumed as mutually independent, mutual information for each signal is zero and thus the Kullback-Liebler convergence is minimized to recover original source signals. A learning algorithm for neural networks as a demixer is derived by mutual information and higher order moments so as to minimize the Kullback-Liebler convergence as the neural networks are learned. We use several image sources to test the purposed approach for computer simulation and obtained simulation results with the separated signals close to the original sources. Future work includes investigation of online learning algorithm for real time blind source separation in practical application.

Appendix I

From the definition of entropy in (5), value of the following entropy is invariant

$$h(X+c) = h(X) \qquad \text{(A-1)}$$

where c is constant. Another useful formula is

$$h(\alpha X) = h(X) + \log|a| \qquad \text{(A-2)}$$

where a is a scaling factor. Consider a probability density function scaled by a

$$f_r(y) = \frac{1}{|a|} f_r\left(\frac{y}{a}\right) \qquad \text{(A-3)}$$

We can rewrite the formula of entropy applying (5)

$$h(Y) = -E[\log f_r(y)] \quad (A\text{-}4)$$
$$= -E\left[\log\left(\frac{1}{|a|} f_r\left(\frac{y}{a}\right)\right)\right]$$
$$= -E\left[\log f_r\left(\frac{y}{a}\right) - \log|a|\right]$$
$$= -E\left[\log f_r\left(\frac{y}{a}\right)\right] + \log|a|$$

By putting $Y = aX$ in (56), finally we obtain

$$h(aX) = -\int_{-\infty}^{\infty} f_X(x)\log f_X(x)dx + \log|a| \quad (A\text{-}5)$$
$$= h(X) + \log|a|$$

Appendix II

Proof of (28)
The mixing matrix W is rewritten by using Laplace's expansion [20] as

$$\det(W) = \sum_{i=1}^{m} w_{ik} A_{ik} \quad (A\text{-}6)$$

where A_{ik} denote ikth cofactor of W. Substituting (A-6) to the left hand side in (26) we have

$$\frac{\partial}{\partial w_{ik}} log|\det(W)| = \frac{1}{|\det(W)|}\frac{\partial}{\partial w_{ik}}|\det(W)| \quad (A\text{-}7)$$
$$= \frac{A_{ik}}{|\det(W)|}$$
$$= (W^{-T})_{ik}$$

Proof of (27)-(30)
First, we calculate the partial derivative with respect to wik for the third moment

$$\frac{\partial}{\partial w_{ik}}(k_{i,3}) = \frac{\partial}{\partial w_{ik}} E(y_i^3) \quad (A\text{-}8)$$
$$= \frac{\partial}{\partial w_{ik}} E((w_{ik}x_k)^3)$$
$$= 3E((w_{ik}x_k)^2)\left(\frac{\partial}{\partial w_{ik}} E((w_{ik}x_{ik}))\right)$$
$$= 3E(y_i^2)x_k$$

and fourth moments

$$\frac{\partial}{\partial w_{ik}}(k_{i,4}) = \frac{\partial}{\partial w_{ik}}(E(y_i^4) - 3(E(y_i^2))^2) \quad (A\text{-}9)$$
$$= \frac{\partial}{\partial w_{ik}}(E(y_i^4))$$
$$= \frac{\partial}{\partial w_{ik}}(E(w_{ik}x_k)^4)$$
$$= 4E((w_{ik}x_k)^3)x_k$$
$$= 4E(y_i^3)x_k$$

By substituting (A-8) and (A-9), we have

$$\frac{\partial}{\partial w_{ik}}(k_{i,3})^2 = 2k_{i,3}\frac{\partial}{\partial w_k}k_{i,3} \quad (A\text{-}10)$$
$$= 2E(y_i^3)3(y_i^2)x_k$$
$$= 6E(y_i^3)(y_i^2)x_k$$

$$\frac{\partial}{\partial w_{ik}}(k_{i,4})^2 = 2k_{i,4}\frac{\partial}{\partial w_k}k_{i,4} \quad (A\text{-}11)$$
$$= 2(E(y_i^4) - 3)(4E(y_i^3)x_k)$$
$$= 8E(y_i^4)E(y_i^3)x_k - 24E(y_i^3)x_k$$

$$\frac{\partial}{\partial w_{ik}}(k_{i,3})^2 k_{i,4} = \frac{\partial}{\partial w_k}k_{i,3}^2 k_{i,4} + k_{i,3}^2\frac{\partial}{\partial w_{ik}}k_{i,4} \quad (A\text{-}12)$$
$$= 6E(y_i^3)E(y_i^2)x_k(E(y_i^4) - 3)$$
$$+ (E(y_i^3))^2 4E(y_i^3))^3 x_k$$
$$= 6E(y_i^4)E(y_i^3)E(y_i^2)x_k$$
$$- 18E(y_i^3)E(y_i^2)x_k + 4E(y_i^3))^3 x_k$$

$$\frac{\partial}{\partial w_{ik}}(k_{i,4})^3 = 3k_{i,4}^2\frac{\partial}{\partial w_k}k_{i,4} \quad (A\text{-}13)$$
$$= 3(E(y_i^4) - 3)^2 4E(y_i^3)x_k$$
$$= 12(E(y_i^4)E(y_i^3))^2 E(y_i^3)x_k$$

Acknowledgment

This research was supported by Ministry of Knowledge and Economy, Republic of Korea under the ITRC(Information Technology Research Center) support program supervised by IITA(Institute for Information Technology Advancement) (IITA-2008-C1090-0801-0004).

References

[1] G. Huang and L. Yang, "A radar anti jamming technology based on blind source separation," *Proc. of Int. Conf. on Signal Processing*, vol. 3, pp. 2021-2024, 2004.

[2] E. Visser and T. W. Lee, "Speech enhancement using blind source separation and two channel energy based speaker detection,", *Proc. of IEEE Int. Conf. on Acoustics, Speech, & Signal Processing*, vol. 1, pp. 884-887, 2003.

[3] Q. Pan and T. Aboulnasr, "A post filter perceptual convolutive blind source separation approach for speech signal," *Proc. of Int. Conf. on Signal Processing*, vol. 1, pp. 327-330, 2004.

[4] D. Kim, H. Choi, and H. Bae, "Acoustic echo cancellation using blind source separation," *IEEE Workshop on Signal Processing Systems*, pp. 241-244, 2003.

[5] C. W. Hesse and C. J. James, "On semi blind

source separation using spatial constraints with applications in EEG analysis," *IEEE Trans. on Biomedical Engineering*, vol. 53, no. 12, pp. 2525-2534, 2006.

[6] K. H. Ting, C. Chang, A. Leung, C. Chan, P. Fung, and F. Chan, "Second-order statistics based blind source separation for artifact correction of short ERP epochs," *IEEE EMBS Asian Pacific Conf. on Biomedical Engineering*, pp. 186-187, 2003.

[7] S. Jayaraman, G. Sitaraman, and R. Seshadri, "Blind source separation of acoustic mixtures using time frequency domain independent component analysis," *Proc. of Int. Conf. on Neural Information Processing*, vol. 3, pp. 1383-1387, 2002.

[8] A. T. Erdogan, "Globally convergent deflationary instantaneous blind source separation algorithm for digital communication signals," *IEEE Trans. on Signal Processing*, vol. 55, no. 5, pp. 2182-2192, 2007.

[9] Q. H. Lin, F. L. Yin, T. M. Mei, and H. Liang, "A blind source separation based method for speech encryption," *IEEE Trans. on Circuits and Systems I*, vol. 53, no. 6, pp. 1320-1328, 2006.

[10] J. Karhunen, E. Oja, L. Wang, R. Vigario, and J. Joutsensalo, "A class of neural networks for independent component analysis," *IEEE Trans. on Neural Networks*, vol. 8, pp. 486-504, 1997.

[11] S. Amari and A. Cichocki, "Adaptive blind signal processing neural network approaches," *Proc. of the IEEE*, vol. 86, no. 10, pp. 2026-2048, 1998.

[12] E. Oja, "From neural network learning to independent components," *Neurocomputing*, vol. 22, pp. 187-199, 1998.

[13] J. F. Cardoso, "Blind signal separation: statistical principles," *Proc. of IEEE*. vol. 86, no. 10, pp. 2009-2025, 1998.

[14] T. W. Lee, M. Girolami, A. J. Bell, and T. J. Sejnowski, "A unifying information theoretic framework for independent component analysis," *Computers & Mathematics with Applications*, vol. 39, pp. 1-21, 2000.

[15] K. Torkkola, "Blind separation of convolved sources based on information maximization," *IEEE Workshop on Neural Networks for signal Processing*, Kyoto, Japan, pp. 423-432, 1996.

[16] S. Amari and A. Cichocki, and H. H. Yang, "A new learning algorithm for blind signals separation," *Advances in Neural Information Processing* 8, MIT Press, pp. 757-763, 1996.

[17] J. F. Cardoso, "Infomax and maximum likelihood for blind separation," *IEEE Letters on Signal Processing*, vol. 4, pp. 112-114, 1997.

[18] J. M. Mendel, *Lessons in estimation theory for signal processing, communications, and control*, Prentice Hall, New Jersey, 1995.

[19] T. Cover and J. Thomas, editors, *Elements of information theory: volume* 1, John Wiley and Sons, New York, 1991.

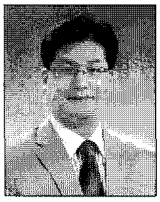
[20] M. G. Kendall and A. Stuart, *Kendall's advanced theory of statistics, volume* 1: *distribution theory*, Oxford University Press, 1998.

[21] P. Comon, "Independent component analysis, a new concept?," *Signal Processing*, vol. 36, pp. 287-314, 1994.

[22] S. Amari, "Natural gradient works efficiently in learning," *Neural Computation*, vol. 10, no. 2, pp. 251-276, 1998.

[23] E. Bingham and A. Hyvärinen, "A fast fixed point algorithm for independent component analysis of complex valued signals," *Int. J. of Neural Systems*, vol. 10, no. 1, pp. 1-8, 2000.

저 자 소 개



Hyun Cheol Cho

He received a B.S. from the Pukyong National University in 1997, an M.S. from the Dong-A University, Korea in 1999, and a Ph.D. from University of Nevada-Reno, USA in 2006. He is currently a post-doctor researcher in the Dept. of Electrical Engineering, Dong-A University. His research interests are in the areas of control systems, stochastic systems, neural networks, and embedded systems.

Tel : 82-51-200-6950

Fax : 82-51-200-7743

E-mail : hyuncho@gmail.com

Kwon Soon Lee

He received a B.S. from Chungnam National University, Korea in 1977, an M.S. from Seoul National University, Korea in 1981, and a Ph. D. in the Dept of Electrical and Computer engineering from Oregon State University, USA in 1990. He is currently worked as a professor in the Dept of Electrical Engineering, Dong-A University. His research interests include all aspects of port automation systems, intelligent control theory, and application of immune algorithm

Tel : 82-51-200-7739

Fax : 82-51-200-7743

E-mail : kslee@dau.ac.kr