ON-LINE BLIND SIGNAL EXTRACTION METHODS EXPLOITING A PRIORI KNOWLEDGE OF THE PREVIOUSLY EXTRACTED SIGNALS

Andrzej CICHOCKI[†], *Ruck THAWONMAS*[†*], *and Shun-ichi AMARI*[†]

[†]Brain Information Processing Group, FRP
The Institute of Chemical and Physical Research (RIKEN)
2-1 Hirosawa, Wako-shi, Saitama 351-01 JAPAN
{cia, ruck, amari}@zoo.riken.go.jp

ABSTRACT

Two alternative neural-network methods are presented which both extract independent source signals one-by-one from a linear mixture of sources when the number of mixed signals is equal to or larger than the number of sources. Both methods exploit the previously extracted source signals as a priori knowledge so as to prevent the same signals from being extracted several times. One method employs a deflation technique which eliminates from the mixture the already extracted signals and another uses a hierarchical neural network which avoids duplicate extraction of source signals by inhibitory synapses between units. Extensive computer simulations confirm the validity and high performance of our methods.

1. INTRODUCTION

Blind source separation can be formulated as the task to recover the unknown sources from the sensor signals described by $\mathbf{x}(t) = \mathbf{A}\mathbf{s}(t)$, where $\mathbf{x}(t)$ is an $n \times 1$ sensor vector, $\mathbf{s}(t)$ is an $m \times 1$ unknown source vector having independent and zero-mean signals, and $\mathbf{A}$ is an $n \times m$ unknown full-rank mixing matrix.

By assuming that the number of sources is known and usually equal to the number of sensors, most of the algorithms in the literature can efficiently perform separation of the source signals in a fully parallel manner [2-5,8,10,13,15]. In practise, the number of sources is unknown and can change rapidly in time, and is usually smaller than the number of sensors, i.e., $m \leq n$ [7]. One possible solution for practical cases is to extract source signals sequentially (one-by-one) [6,9-12,18,19]. This solution requires use of two techniques: one for extracting a single source signal form the mixture and another for preventing duplicate extraction of the same sources by next processing units.

To extract a single source signal, methods for blind equalization or deconvolution problems [17] can be used, as done in [6,9,11,12,14,18,19]. Namely, extraction of an independent source signal can be achieved by maximizing (and/or minimizing) the fourth order cumulants $\kappa_4(y_1)$ subject to certain constraints. To prevent duplicate extraction, an adaption of the orthogonal Schur eigenvalue deflation technique was used in [9]. This technique is, however, not suited for on-line, real-time applications due to its rather high complexity. In [11], the hierarchical orthogonalization technique [16] was used. However, it is rather difficult to choose proper values for the coefficients corresponding to the orthogonalizing feedback terms, unless a priori knowledge of the kurtosis of sources signals is known.

In this paper, we present two alternative neural-network methods for extracting source signals on-line when $m \leq n$. One method (in Section 2) employs a deflation technique for eliminating from the mixture thc already extracted signals while another method (in Section 3) uses a hierarchical neural network which avoids duplicate extraction of the source signals by inhibitory synapses between units. One salient feature of these methods is that they can extract first the most "interesting" signals, those most deviated from Gaussian signals. In addition, the learning adaptive algorithms in use are purely local, biological plausible and often simpler than algorithms for blind signal separation.

2. ON-LINE EXTRACTION AND DEFLATION LEARNING ALGORITHMS

Let us consider a single processing unit (see Fig. 1.a) $y_1 = \mathbf{w}_1^T \mathbf{x}_1 = \sum_{j=1}^{n} w_{1j} x_{1j}$, where $\mathbf{x}_1 = [x_{11}, x_{12}, \ldots, x_{1n}]^T$ is either the mixing signals $\mathbf{x} = \mathbf{A}\mathbf{s}$ or the prewhitened version of $\mathbf{x}$. Prewhitening or decorrelation

[*]This author is supported in part by the Special Postdoctoral Researchers Program of RIKEN.

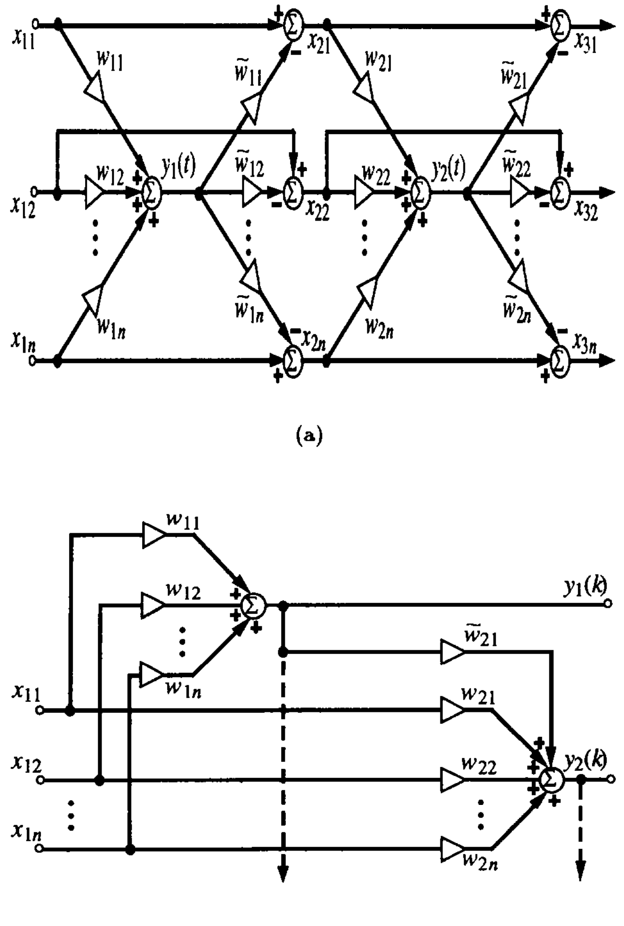


Figure 1: The architectures of the extraction-deflation neural network (a) and the hierarchical neural network (b).

of $\mathbf{x}$ so that $\mathbf{R}_{\mathbf{x}_1\mathbf{x}_1} = \mathbf{I}$ [1,19] improves the recovering performance for the cases where mixing matrixes $\mathbf{A}$ are ill-conditioned or where additive noises exist at the sensors. The unit successfully extracts a source signal if $\mathbf{w}_1(t) = \mathbf{w}_{1*}$ satisfy the relation $\mathbf{w}_{1*}^T\mathbf{A} = \mathbf{e}_k$, where $\mathbf{e}_k$ denotes the k-th column of the $n \times n$ identity matrix $\mathbf{I}$.

A possible loss (contrast) function can be formulated as [18]:

$$\mathcal{J}_1(\mathbf{w}_1) = -\frac{1}{4}|\kappa_4(y_1)|, \qquad (1)$$

where $\kappa_4(y_1)$ is normalized kurtosis defined as $\kappa_4(y_1) = \frac{E\left[y_1^4\right]}{E^2\left[y_1^2\right]} - 3$. Minimization of the loss function (1) leads to a simple learning rule:

$$\frac{d\mathbf{w}_1}{dt} = -\mu_1(t)f_1(y_1(t))\mathbf{x}_1(t), \qquad (2)$$

or

$$\mathbf{w}_1(t+1) = \mathbf{w}_1(t) - \mu_1(t)f_1(y_1(t))\mathbf{x}_1(t),$$

where $\mu_1(t) > 0$ is a learning rate and $f(y_1(t)) = sign\left[\kappa_4(y_1(t))\right]\left[y_1(t) - \frac{m_2\left(y_1(t)\right)}{m_4\left(y_1(t)\right)}y_1^3(t)\right]\frac{m_4\left(y_1(t)\right)}{m_2^3\left(y_1(t)\right)}$, with $m_p(t) \overset{\text{def}}{=} E\left[|y_1(t)|^p\right]$, $p = 2,\ 4$. The high order moments m_2, m_4 and the sign of kurtosis κ_4 can be estimated on-line using the following averaging formula:

$$\frac{dm_p(y_k)}{dt} = \mu\left[-m_p\left(y_k(t)\right) + |y_k(t)|^p\right]. \qquad (3)$$

After successful extraction of the first source signal $y_1(t) \sim s_j(t)$ $(j \in \{1, \ldots, n\})$, we can apply a deflation procedure which removes previously extracted signals from the mixture. This means that we are looking for such an on-line linear transformation given by (see Fig. 1.a)

$$\mathbf{x}_{k+1}(t) = \mathbf{x}_k(t) - \tilde{\mathbf{w}}_k(t)y_k(t) \qquad k = 1, 2, \ldots \quad (4)$$

which ensures minimization of the generalized energy (loss) function

$$\tilde{\mathcal{J}}_k(\tilde{\mathbf{w}}_k) = \frac{1}{2}\left\|\mathbf{x}_{k+1}\right\|^2, \qquad (5)$$

where $y_k = \mathbf{w}_k^T\mathbf{x}_k$,

$$\frac{d\mathbf{w}_k}{dt} = -\mu_k(t)f_k(y_k(t))\mathbf{x}_k(t), \qquad (6)$$

and $f_k\left(y_k(t)\right) = sign\left[\kappa_4(y_k(t))\right]\ \left[y_k(t) - \frac{m_2\left(y_k(t)\right)}{m_4\left(y_k(t)\right)}y_k^3(t)\right]$ $\frac{m_4\left(y_k(t)\right)}{m_2^3\left(y_k(t)\right)}$. The last term m_4/m_2^3 can be absorbed by learning rate $\mu_k(t)$ since it is always positive. Minimization of the above defined loss function leads to a simple learning rule:

$$\frac{d\tilde{\mathbf{w}}_k}{dt} = \tilde{\mu}_k(t)y_k(t)\mathbf{x}_{k+1}(t) \qquad k = 1, 2, \ldots. \quad (7)$$

The procedure can be continued until all of the estimated source signals are recovered, i.e., until the amplitude of each signal $\mathbf{x}_{k+1,i}$ is below a given threshold. This means that it is not necessary to know the number of source signals in advance.

3. HIERARCHICAL NEURAL NETWORK

Another method different from the deflation procedure described above is to use a hierarchical neural network as shown in Fig. 1.b. The neural model is described by a simple set of equations:

$$\begin{aligned} y_k &= \sum_{j=1}^{n} w_{kj}(t)x_{1j} + \mathrm{c}(t)\sum_{i=1}^{k-1} \tilde{w}_{ki}y_i \\ &= \mathbf{w}_k^T\mathbf{x}_1 + \mathrm{c}(t)\tilde{\mathbf{w}}_k^T\mathbf{y}, \end{aligned} \tag{8}$$

where $\tilde{\mathbf{w}}_k = [\tilde{w}_{k,1}, \tilde{w}_{k,2}, \dots, \tilde{w}_{k,k-1}, 0, \dots, 0]^T$, $\mathbf{y} = [y_1, y_2, \dots, y_n]^T$, and $\mathrm{c}(t)$ is a scaling factor. As done in the previous section, for extraction of a single source signal we employ the following loss (cost) function:

$$\mathcal{J}_k(\mathbf{w}_k) = -\frac{1}{4}|\kappa_4(y_k)|. \tag{9}$$

Here, however, to ensure that the extracted signal y_k is different from the previously extracted signals y_j, where $j < k$, we introduce an additional loss function:

$$\tilde{\mathcal{J}}_k(\mathbf{w}_k, \tilde{\mathbf{w}}_k) = \lambda \sum_{j=1}^{k-1} \left(E[y_j(t)y_k(t)]\right)^2, \tag{10}$$

where $\lambda > 0$ is a penalty parameter. This loss function contributes non-zero penalties when the outputs of the jth and kth units, i.e., y_j and y_k, are correlated.

Applying a standard stochastic gradient descent procedure to (9)+(10), we obtain on-line local adaptive learning rules:

$$\frac{d\mathbf{w}_k}{dt} = -\mu_k(t)f_k(y_k(t))\mathbf{x}_1(t) \tag{11}$$

where $\mu_k > 0$, and

$$\begin{aligned} f_k(y_k(t)) &= sign[\kappa_4(y_k(t))]\,[y_k(t) - \frac{m_2(y_k(t))}{m_4(y_k(t))}y_k^3(t)] \\ &\quad + \lambda\sum_{j=1}^{k-1} r_{kj}(t)y_j(t) \end{aligned} \tag{12}$$

with $r_{kj}(t) \overset{\text{def}}{=} E[y_k(t)y_j(t)]$. As with m_p, the correlation r_{kj} can be estimated on-line using the following averaging formula:

$$\frac{dr_{kj}}{dt} = \mu[-r_{kj} + y_k(t)y_j(t)]. \tag{13}$$

The role of the lateral synaptic weights $\tilde{w}_{kj}$ is devoted to preventing the previously extracted signals y_j, where $j < k$, from being redundantly extracted. The learning rule is, therefore, derived by applying a standard gradient descent procedure to (10). The resulting learning rule is as follows.

$$\frac{d\tilde{w}_{kj}}{dt} = -\tilde{\mu}_k(t)[r_{kj}(t)y_k(t)y_j(t) + \alpha h(\tilde{w}_{kj})] \qquad (j < k) \tag{14}$$

where $\tilde{\mu}_k$ and $\alpha > 0$, and $h(w)$ is a non-linear odd function, such as $h(w) = |w|^p \mathrm{sign}(w), p = 1, 3, 5, \dots$ The weight decay term $\alpha h(\tilde{w}_{kj})$ plays a role of a forgetting term, namely, it forces to zero those weights not receiving sufficient reinforcement.

4. COMPUTER SIMULATIONS

We confirmed the validity and performance of our methods using extensive computer simulations for a variety of problems.

Below, due to limit of space, we only present an illustrative example of typical results from the hierarchical neural network method experimented with three binary 512×512 images. These results were obtained when we initialized all the weights such that they had random values in the range -0.1 and 0.1, used the fixed learning rates of 0.001, and started the kth extraction unit at time $(k-1)2500$. Five linear mixtures of the images were generated by multiplying the source signal vector with the randomly chosen mixing matrix $\mathbf{A}$ =

$$\begin{pmatrix} -0.9846 & -0.8609 & 0.4024 \\ -0.2332 & 0.6923 & 0.8206 \\ -0.4557 & 0.0539 & 0.4239 \\ -0.1650 & -0.8161 & -0.8751 \\ 0.6735 & 0.8078 & -0.9051 \end{pmatrix}.$$

Fig. 2 shows the original images, the mixed images and the extracted images in (a), (b) and (c), respectively. Visual comparison of Figs. 2.a and 2.c confirms that the source signals were successfully extracted, but subject to un-determinacy of the order and the sign of extracted signals.

We note here that the number of active sources in the mixed signals was not known to the system. In the above experiment, at the 4th extraction unit, the correlation between the outputs of the 2nd and 4th units was high, e.g., $r_{42}(10000)$ = -0.8290, and never converged to zero. This implied that all of the active sources had been successfully extracted by the first three extraction units. We could, therefore, terminate the extraction process and discard the result at the 4th unit.

5. CONCLUSIONS

We have presented two alternative neural-network methods for on-line blind signal extraction. Our approach has the following features: It uses a simple cost function (absolute value of normalized kurtosis) without any constraints. From this cost function, simple adaptive nonlinear functions are derived. These nonlinear functions change their shapes during the learning process. Moreover, the proposed algorithms are able to extract signals both sub-Gaussian and super-Gaussian. The developed learning algorithms are purely local and are biologically plausible; they could be considered as a generalization or extension of Hebbian/anti-Hebbian rules. The proposed methodology can be extended to multi-channel blind signal deconvolution or generalized to complex-valued signals.

6. REFERENCES

[1] S. Amari, "Neural theory of association and concept-formulation", *Biological Cybernetics.*, Vol. 26, 1977, pp. 175-185.

[2] S. Amari, "Natural gradient works efficiently in learning", *Neural Computation*, 1997, (in print).

[3] S. Amari, A. Cichocki, and H.H. Yang, "A new learning algorithm for blind signal separation", *Proc. NIPS 95*, MIT Press, Vol. 8, 1996, pp. 757-763.

[4] J.-F. Cardoso and B. Laheld, "Equivariant adaptive source separation", *IEEE Trans. on Signal Processing*, Vol. 44, No. 12, Dec 1996, pp. 3017-3030.

[5] A. Cichocki, S. Amari, M. Adachi, and W. Kasprzak, "Self-adaptive neural networks for blind separation of sources", *Proc. ISCAS 96*, May 1996, Vol. II, pp. 157-160.

[6] A. Cichocki, S. Amari and R. Thawonmas, "Blind signal extraction using self adaptive non-linear hebbian learning rule", *NOLTA'96*, October 1996, pp. 377-380.

[7] A. Cichocki, W. Kasprzak and S. Amari, "Neural network approach to blind separation of enhancement of images", *Proc. EUSIPCO 96*, September 1996, Vol. I, pp. 579-582.

[8] A. Cichocki and R. Unbehauen, "Robust neural networks with on-line learning for blind identification and blind separation of sources", *IEEE Trans. Circuits and Systems-I*, Vol. 43, Nov. 1996, pp. 894-906.

[9] N. Delfosse and P. Loubaton, "Adaptive blind separation of independent sources: A deflation approach", *Signal Processing*, Vol. 45, 1995, pp. 59-83.

[10] M. Girolami and C. Fyfe, "Blind separation of sources using exploratory projection pursuit networks", *Proc. Int. Conf. on Eng. Applications of Neural Networks*, 1996, pp. 249-252.

[11] A. Hyvärinen and E. Oja, "Simple neuron models for independent component analysis", *Journal on Neural Systems*, Vol. 7, No. 6, 1996, pp. 671-687.

[12] A. Hyvärinen, "A family of fixed-point algorithms for independent component analysis", *Proc. ICASSP 97*, April 1997, Vol. V, pp. 3917-3920.

[13] C. Jutten and J. Herault, "Blind separation of sources, Part I: An adaptive algorithm based on neuromimetic architecture", *Signal Processing*, Vol. 24, 1991, pp. 1-20.

[14] Z. Malouche and O. Macchi, "Extended anti-hebbian adaptation for unsupervised source extraction", *Proc. ICASSP 96*, May 1996, pp. 1664-1667.

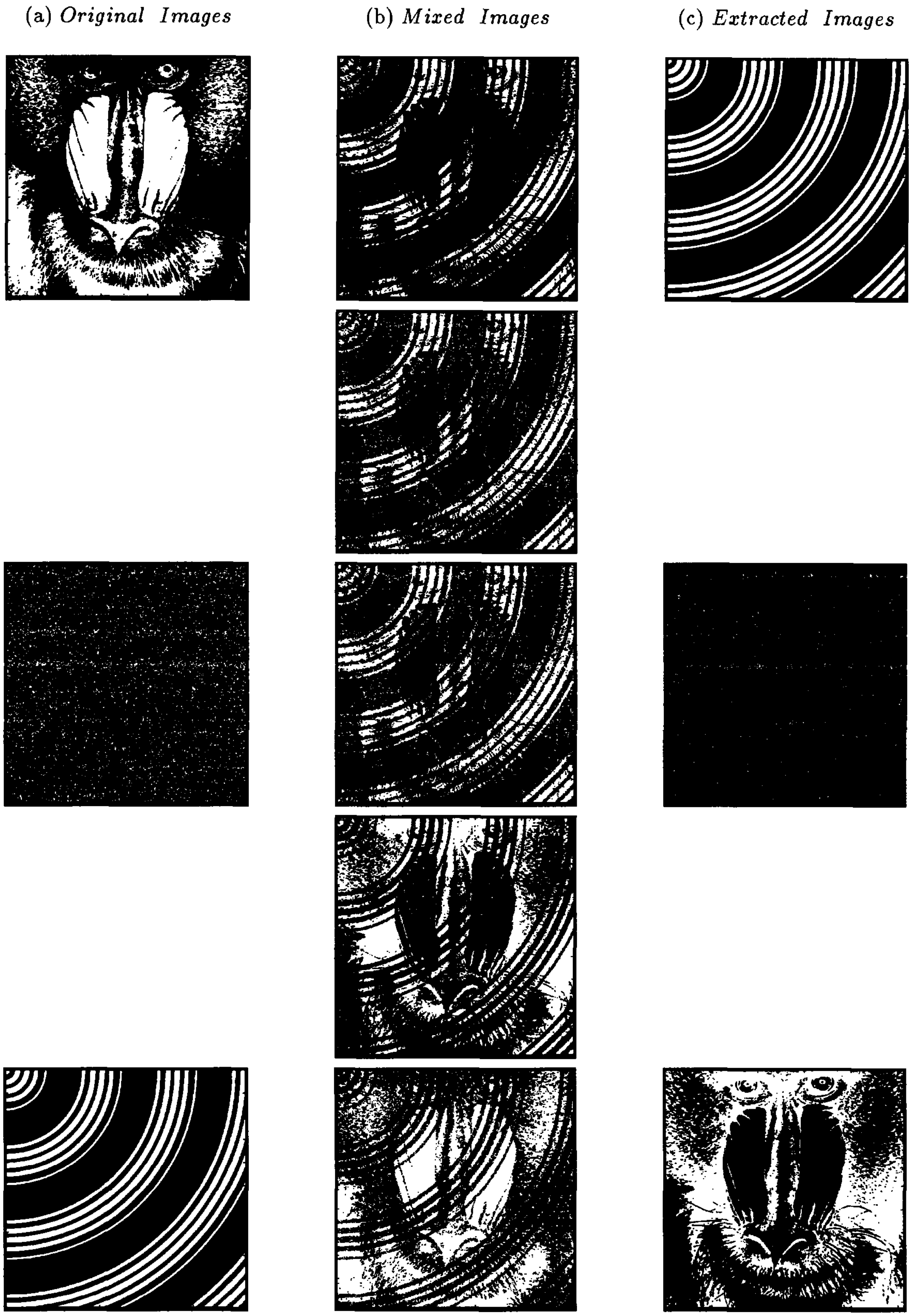
[15] E. Oja and J. Karhunen, "Signal separation by nonlinear hebbian", *Computational Intelligence - a Dynamic System Perspective*, pp. 83-97, IEEE Press, New York, 1995.

[16] T. Sanger, "Optimal unsupervised learning in a single-layered linear feedforward network", *Neural Networks*, Vol. 2, 1989, pp. 459-473.

[17] O. Shalvi and E. Weinstein, "Universal method for blind deconvolution", *Blind Deconvolution*, S. Haykin, Ed., pp. 121-180, Prentice-Hall, Inc., Englewood Cliffs, New Jersey, 1994.

[18] R. Thawonmas and A. Cichocki, "Blind extraction of source signals with specified stochastic features", *Proc. ICASSP 97*, April 1997, Vol. IV, pp. 3353-3356.

[19] A. Cichocki, R. Thawonmas and S. Amari, "Dual cascade networks for blind signal extraction", *Proc. ICNN 97*, June 1997, Vol. IV, pp. 2135-2140.



(a) *Original Images*

(b) *Mixed Images*

(c) *Extracted Images*

Figure 2: Typical results of extraction of three binary 512×512 images received at five sensors.