LINEAR MULTILAYER ICA ALGORITHM INTEGRATING SMALL LOCAL MODULES

Yoshitatsu Matsuda and Kazunori Yamaguchi

Kazunori Yamaguchi Laboratory,
Department of General Systems Studies,
Graduate School of Arts and Sciences, The University of Tokyo,
3-8-1, Komaba, Meguro-ku, Tokyo, 153-8902, *Japan*.
{matsuda,yamaguch}@graco.c.u-tokyo.ac.jp

ABSTRACT

In this paper, the linear (feed-forward) multilayer ICA algorithm is proposed for the blind separation of high-dimensional mixed signals. There are two main phases in each layer. One is the local ICA phase, where the mixed signals are divided into small local modules and a simple ICA is applied to each module. Another is the mapping phase, where the locally-separated signals are arranged as a line so that the higher correlated signals are nearer. By repetition of these two phase, this multilayer ICA algorithm can find all the (global) independent components through only the ICA processing on local modules. Some numerical experiments on artificial data and natural scenes show the validity of this algorithm, and verify that it is more efficient than the standard fast ICA algorithm for "locally-biased" and high-dimensional observed signals such as natural scenes.

1. INTRODUCTION

Independent component analysis (ICA) is a recently developing method in the fields of signal processing and artificial neural networks, and has been shown to be quite useful for the blind separation problem (Jutten & Herault, 1991; Comon, 1994; Bell & Sejnowski, 1995; Amari & Cichocki, 1998). The linear ICA is formalized as follows. Let $\boldsymbol{s}$ and $\boldsymbol{A}$ are N-dimensional original signals and $N \times N$ mixing matrix. Then, the observed signals $\boldsymbol{x}$ are defined as

$$\boldsymbol{x} = \boldsymbol{A}\boldsymbol{s}. \tag{1}$$

The purpose is to find out $\boldsymbol{A}$ (or the inverse $\boldsymbol{W}$) when the observed (mixed) signals only are given. In other words, ICA separates the observed signals into the original signals blindly as follows:

$$\hat{\boldsymbol{s}} = \boldsymbol{W}\boldsymbol{x}, \tag{2}$$

where $\hat{\boldsymbol{s}}$ is the estimate of the original signals. This is a typical ill-conditioned problem, but ICA can solve it by assuming that the original signals are generated according to independent and non-gaussian probability distributions. In general, the ICA algorithms find out the $\boldsymbol{W}$ maximizing a criterion such as the higher order statistics (e.g the kurtosis) of every component of $\hat{\boldsymbol{s}}$. That is, the ICA algorithms can be regarded as an optimization method of such criteria. Some efficient algorithms for this optimization problem have been proposed, for example, the fast ICA algorithm (Hyvärinen & Oja, 1997; Hyvärinen, 1999) and the natural gradient algorithm (Amari, 1998; Amari & Cichocki, 1998.)

Now, suppose that quite high-dimensional observed signals (namely, N is quite large) are given such as video images. In this case, even the efficient algorithms are not much useful because they have to find out all the N^2 components of $\boldsymbol{W}$. The devision of the observed signals is an easy method to avoid this problem. For the video images, they are divided into small patches (e.g. the squares of 8×8 pixels) and each patch are analyzed by ICA (e.g. see Bell & Sejnowski (1997) and van Hateren & Ruderman (1998)). This method is quite simple, but include some difficulties. The most intractable problem is that this division scheme inevitably neglects global independent components spreading over multiple patches. This problem can be alleviated by using the larger patches, but the computational costs become heavier. In this paper, we propose the linear multilayer ICA algorithm integrating local modules (the generalization of patches). This algorithm can find out global independent components by repetition of ICA on local modules while updating the mapping (or arrangement) of the signals. Though it extracts only the linear independent components unlike the other multilayer ICA algorithms (e.g. Hyvärinen & Hoyer (2000)), it is expected to be more efficient than the standard linear ICA algorithms when N is large.

This paper is organized as follows. In Section 2, the algorithm is described in detail. Section 3 shows some numerical results in the blind separation of uniformly distributed random variables and the extraction of edge detectors from natural scenes. In addition, the validity of this algorithm is discussed based on these results. Lastly, this paper is concluded in Section 4.

2. ALGORITHM

2.1. overview

The basic idea of our multilayer ICA algorithm is quite simple. That is, the originally independent signals (among which the (not only second- but higher-order) correlations are close to zero) need not be included in the same local module. In other words, the highly correlated signals must be near each other so that they can be allocated to the same module. According to this simple idea, our algorithm makes the signals form a mapping where the correlated signals are nearer in "the mapping phase". The optimal localization is achieved by this phase. Then, the signals are divided into small local modules, and the simple ICA algorithm is applied to each module in "the local ICA phase".

In addition, the global independent components spreading over multiple modules are expected to be found out by repetition of these two phase. Fig. 1 illustrates the most ideal case where the number of mixed signals is 9 and the size of local modules is 3. It is easy to calculate the minimal number of layers acquiring all the global independent components. It is $\log_n N$ where n is the size of local modules and N is the number of signals. Note that it holds in the most ideal case only and such regular separations can not be expected in the practical case. Nevertheless this estimation suggests the validity of the linear multilayer ICA algorithm.

2.2. mapping phase

In what follows, the linear multilayer ICA algorithm is described formally. Let $\boldsymbol{X} = (x_{ik})$ be an $N \times M$ given signal matrix where M is the number of samples. Next, a higher-order covariance between the i-th and j-th signals, α_{ij}, is defined as

$$\alpha_{ij} = \sum_{k=1}^{M} x_{ik}^2 x_{jk}^2 - \sum_k x_{ik}^2 \sum_k x_{jk}^2, \tag{3}$$

which is the second order covariance between the squared signals. Then, the following criterion for the mapping formation is defined:

$$E = \sum_{i,j} \alpha_{ij}^2 d_{ij}, \tag{4}$$

where d_{ij} is the square Euclidean distance between the i-th and j-th signals in a $1 \times N$ array. In the mapping phase, we arrange the signals in the $1 \times N$ array so that E is minimized. Fast Simulated Annealing (SA) is used for the optimization (Szu & Hartley, 1987)[1]. Note that such an arrangement can be represented as an $N \times N$ permutation matrix $\boldsymbol{P}$ because a one-dimensional array is used. In other words, the output signal matrix from the mapping phase, $\boldsymbol{X}^{\text{map}}$, is calculated as

$$\boldsymbol{X}^{\text{map}} = \boldsymbol{P}\boldsymbol{X}. \tag{5}$$

[1] Although fast SA is employed in this paper for simplicity, there exist more efficient algorithms (e.g the analytical procedure (Cox & Cox, 1994) and the self organization (Goodhill & Sejnowski, 1997; Matsuda & Yamaguch, 2002)) for optimizing such a criterion as Eq. (4).

2.3. local ICA phase

In the local ICA phase, the given signal matrix are divided every n rows. The divided signal matrix for the p-th local module, $\boldsymbol{X}_p^{\text{modu}}$, is given as

$$\boldsymbol{X}_p^{\text{modu}} = \begin{pmatrix} x_{np-n+1,1} & \cdots & x_{np-n+1,M} \\ \cdots & \cdots & \cdots \\ x_{np,1} & \cdots & x_{np,M} \end{pmatrix}. \tag{6}$$

Then, the $n \times n$ inverse (separation) matrix $\boldsymbol{W}_p^{\text{modu}}$ is calculated for each $\boldsymbol{X}_p^{\text{modu}}$. This simple ICA is carried out by the fast ICA algorithm (Hyvärinen & Oja, 1997; Hyvärinen, 1999). By gathering every $\boldsymbol{W}_p^{\text{modu}}$, the following matrix $\boldsymbol{W}^{\text{local}}$ is obtained:

$$\boldsymbol{W}^{\text{local}} = \begin{pmatrix} \boldsymbol{W}_1^{\text{modu}} & \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{W}_2^{\text{modu}} & \boldsymbol{0} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} & \cdots & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{W}_{\frac{N}{n}}^{\text{modu}} \end{pmatrix}, \tag{7}$$

and the output signal matrix from the local ICA phase is given as

$$\boldsymbol{X}^{\text{local}} = \boldsymbol{W}^{\text{local}}\boldsymbol{X}. \tag{8}$$

2.4. complete algorithm

The linear multilayer ICA algorithm is given completely as follows.

1. Initially Settings: Let $\boldsymbol{X}$ and $\boldsymbol{W}$ be the observed signal matrix and the $N \times N$ identity matrix, respectively.
2. Mapping Phase: Calculate the optimal permutation matrix $\boldsymbol{P}$ minimizing Eq. (4). Then, let
$$\boldsymbol{X} := \boldsymbol{P}\boldsymbol{X} \tag{9}$$
and
$$\boldsymbol{W} := \boldsymbol{P}\boldsymbol{W}. \tag{10}$$
3. Local ICA Phase: Divide $\boldsymbol{X}$ into local modules, and calculate $\boldsymbol{W}^{\text{local}}$. Then, let
$$\boldsymbol{X} := \boldsymbol{W}^{\text{local}}\boldsymbol{X} \tag{11}$$
and
$$\boldsymbol{W} := \boldsymbol{W}^{\text{local}}\boldsymbol{W}. \tag{12}$$

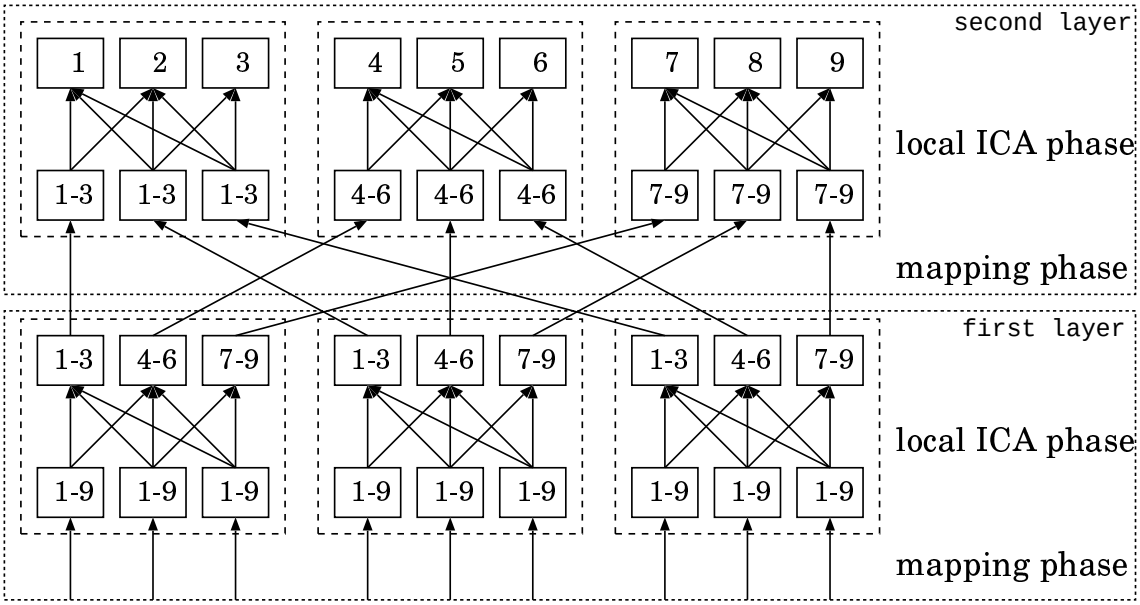


Fig. 1. The ideal case of the linear multilayer ICA algorithm: This illustrates the most ideal case in a linear blind separation problem. The number of mixed (and original) signals and the size of local modules are nine and three, respectively. Here, the mapping phase in the first layer is omitted. In the first local ICA phase, the mixed signals of nine original ones are separated into the locally-separated signals of three original ones. Then, the signals of the same three signals are arranged nearer in the second mapping phase. Lastly, the original nine signals are extracted in the second local ICA phase.

4. Go to Next Layer: Terminate the learning process if a termination condition is satisfied. Otherwise, return to STEP 2.

Because this algorithm is feed-forward, it needs only the current $\boldsymbol{W}$ and $\boldsymbol{X}$. So, the required memory space is constant even if the number of layers increases.

3. RESULTS

In order to verify the validity of the linear multilayer ICA algorithm, we carried out two numerical experiments. One is the blind separation of the mixed signals of uniformly distributed random variables. Another is the extraction of edge detectors from natural scenes.

3.1. blind separation of uniformly distributed random variables

3.1.1. settings

32 uniformly random variables were generated for 10000 times as the source signals. That is, the dimensional number of signals N and the sampling number M were 32 and 10000, respectively. Then, the mixed signals $\boldsymbol{X}$ were constructed by a 32×32 mixing matrix $\boldsymbol{A}$ according to Eq. (1). In order to simulate the "locally-biased" observed signals, locally-biased mixing matrices were used. An l-th locally-biased mixing matrix $\boldsymbol{A}_l^{\text{bias}}$ is given as a sparse matrix whose diagonal and the $(l-1)$ right succeeding components are not zero. Here, each row is regarded as a circular chain. For example, $\boldsymbol{A}_2^{\text{bias}}$ is given as

$$\boldsymbol{A}_2^{\text{bias}} = \begin{pmatrix} a_{1,1} & a_{1,2} & \mathbf{0} & 0 & 0 \\ \mathbf{0} & \cdots & \cdots & \cdots & \mathbf{0} \\ 0 & 0 & \mathbf{0} & a_{N-1,N-1} & a_{N-1,N} \\ a_{N,1} & 0 & \mathbf{0} & 0 & a_{N,N} \end{pmatrix}, \tag{13}$$

where each non-zero a_{ij} is given as a random variable. Note that an N-th locally-biased matrix is a normal $N \times N$ random matrix. In this paper, mixed signals constructed by such a normal random matrix are called "totally-mixed" ones. $\boldsymbol{A}_2^{\text{bias}}$, $\boldsymbol{A}_4^{\text{bias}}$, $\boldsymbol{A}_6^{\text{bias}}$, and $\boldsymbol{A}_{32}^{\text{bias}}$ (totally-mixed) were employed in this experiment.

In the linear multilayer ICA algorithm, the size of local modules n was set 4 ($\ll N = 32$). The algorithm was terminated if the number of layers reached 100. In other words, the max number of layers was 100. The fast SA parameters in the mapping phase were given as follows: Move set: pairwise interchanges. Initial temperature t_0: 100.0 $\times$ the mean energy difference over N^2 moves. Cooling schedule: The temperature at the step T is $\frac{t_0}{T}$. Stopping

criterion: $T = 10N^2$. In the fast ICA of the local ICA phase, $g(u) = u^3$ (the kurtosis criterion) and the deflation approach were employed.

3.1.2. results

Fig. 2 shows the decreasing curves of the separation error along the number of layers. The error is measured as the square root of the summation over all the squared components of the difference between $\boldsymbol{WA}$ and the most similar permutation matrix. In order to alleviate the probabilistic fluctuation, its median (and quartiles) over 50 runs is exhibited. The results of PCA and the standard fast ICA algorithm are also displayed as horizontal dotted lines for comparison.

Totally-mixed signals ($l = 32$): The decreasing curve of the error with the upper and lower quartiles (namely, the 75th and the 25th percentile) are shown in Fig. 2-(a). It verifies that the linear multilayer ICA algorithm could find out all the (global) independent components rather accurately and robustly. In addition, it suggests that the error decreases exponentially according to the number of layers.

Locally-biased mixed signals ($l = 2, 4, 6$): Fig. 2-(b) shows four decreasing curves for l = 2, 4, 6, and 32. The error was smaller and converged faster for smaller l.

Calculation time: When $N = 32$, the standard fast ICA required 17.1 seconds as the average calculation time with a Pentium III 1GHz processor. On the other hand, the linear multilayer ICA spent 13.7 seconds on the calculation (including the mapping phase and the local ICA phase) at each layer. In addition, fast ICA required 0.41 seconds for $N = 4$ (namely, for each local module of $m = 4$).

3.2. extraction of edge detectors

It has been well-known that various local edge detectors can be extracted from natural scenes by the standard ICA algorithm (Bell & Sejnowski, 1997; van Hateren & Ruderman, 1998). Here, the multilayer ICA algorithm was applied to the same problem.

The resulting basis vectors are shown in Fig. 3. Fig. 3-(a), (b), and (c) correspond to the results at the 1st layer, 5th layer, 20th layer, respectively. Fig. 3-(d) displays the ones generated by the standard fast ICA algorithm. It seems that many edge detectors were extracted already at the 5th layer (Fig. 3-(b)) and the result at the 20th layer is qualitatively similar to that of the standard ICA (Fig. 3-(c) and (d)).

Regarding the calculation time, the linear multilayer ICA algorithm spent 12300 seconds on the total computation of the 20 layers (so, it required 620 seconds on average at each layer), while the standard ICA algorithm needed 13200 seconds in total.

3.3. discussions

Firstly, the results in Fig. 2-(a) verify that the linear multilayer ICA algorithm always finds out all the independent components by applying ICA to the small local modules only. Though the accuracy of the separation was slightly worse than the standard ICA, it was close to zero. In addition, the separation succeeded under any initial condition (the original signals and the mixing matrix) and parameter setting (N, n, and l). It is interesting that the quite simple scheme (repetition of the mapping formation according to the higher-order correlations and the simple ICA on the small local modules) can solve the general (and global) ICA problem robustly.

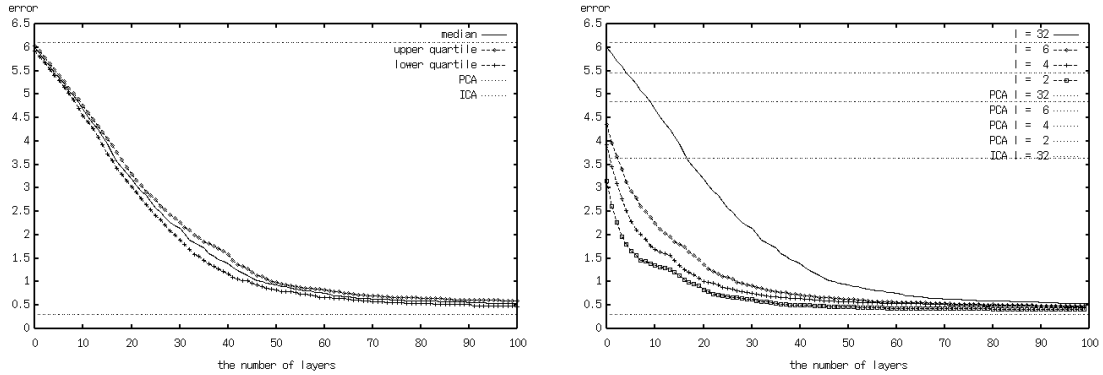
Secondly, Fig. 2-(b) indicates that the error converged faster (within about 10-20 layers) for locally-biased (smaller l) signals. Unfortunately, in this blind separation problem of such locally-biased signals, the multilayer linear ICA algorithm was poorer in efficiency than the standard algorithm in respect to the calculation time because of the high computational costs of the fast SA algorithm. Nevertheless, this faster convergence suggests that the multilayer algorithm is suitable for such signals if a more efficient algorithm is employed in the mapping phase.

Thirdly, Fig. 3 illustrates that the extraction of edge detectors from natural scenes was approximately acquired through only several (about 5) layers. In addition, the linear multilayer ICA algorithm needed less calculation time than the standard ICA even for the total 20 layers, whose result (Fig. 3-(c)) is quite similar to the standard one (Fig. 3-(d)). It is probably because the costs of fast SA become relatively low if the dimensional number of signals is high (e.g. 144). These results strongly verify that the linear multilayer ICA algorithm is more efficient than the standard fast ICA algorithm for locally-biased and high-dimensional observed signals such as the image patches from natural scenes.

4. CONCLUSION

In this paper, we proposed the linear multilayer ICA algorithm integrating local modules. The numerical experiments showed that this algorithm can find out all the independent components (including the global components spreading over multiple modules). In addition, they shows that it is more efficient than the standard ICA algorithm for locally-biased mixed signals such as natural scenes.

Future work is as follows. First, the optimization in the mapping phase must be improved. Fast SA is simple but not efficient. A more efficient algorithm is required. Next, this algorithm may be applied to quite large-scale signals



(a) totally-mixed signals. (b) locally-biased mixed signals.

Fig. 2. The blind separation by the linear multilayer ICA algorithm: This shows the decreasing curves of the separation error along the number of layers. The upper horizontal lines and the most lower one are the results of PCA and the standard fast ICA, respectively. Since the results of PCA depended on l, all the PCA lines are displayed. Regarding the standard ICA, only the line for $l = 32$ is displayed because their results were almost constant regardless of l. (a): It shows the decreasing curve of the separation error for the totally-mixed signals ($l = 32$). In addition, the upper and lower quartiles are shown as the dotted lines. (b): It shows four PCA lines and four curves for l = 2, 4, 6, and 32 from the lower.

such as 256×256 images. Lastly, we want to apply some nonlinear operators to each layer. This approach may lead to a generalization of nonlinear ICA algorithms. For example, letting the components in the second layer be squared, this algorithm seems to be similar to Hyvärinen and Hoyer's network (Hyvärinen & Hoyer, 2000).

5. REFERENCES

[1] Amari, S. (1998). Natural gradient works efficiently. *Neural Computation,* 10, 251-276.

[2] Amari, S. & Cichocki, A. (1998). Adaptive blind signal processing: Neural network approaches. *Proceedings of the IEEE,* 86, 2026-2048.

[3] Bell, A. J. & Sejnowski, T. J. (1995). An information-maximization approach to blind separation and blind deconvolution. *Neural Computation,* 7, 1129-1159.

[4] Bell, A. J. & Sejnowski, T. J. (1997). The "independent components" of natural scenes are edge filters. *Vision Research,* 37, 3327-3338.

[5] Comon, P. (1994). Independent component analysis - a new concept? *Signal Processing,* 36, 287-314.

[6] Cox, T. F. & Cox, M. A. A. (1994). *Multidimensional scaling.* London: Chapman & Hall.

[7] Goodhill, G. J. & Sejnowski, T. (1997). A unifying objective function for topographic mappings. *Neural Computation,* 9, 1291-1303.

[8] Hyvärinen, A. (1999). Fast and robust fixed-point algorithms for independent component analysis. *IEEE Transactions on Neural Networks,* 10, 626-634.

[9] Hyvärinen, A. & Hoyer, P. O. (2000). Emergence of phase and shift invariant features by decomposition of natural images into independent feature subspaces. *Neural Computation,* 12, 1705-1720.

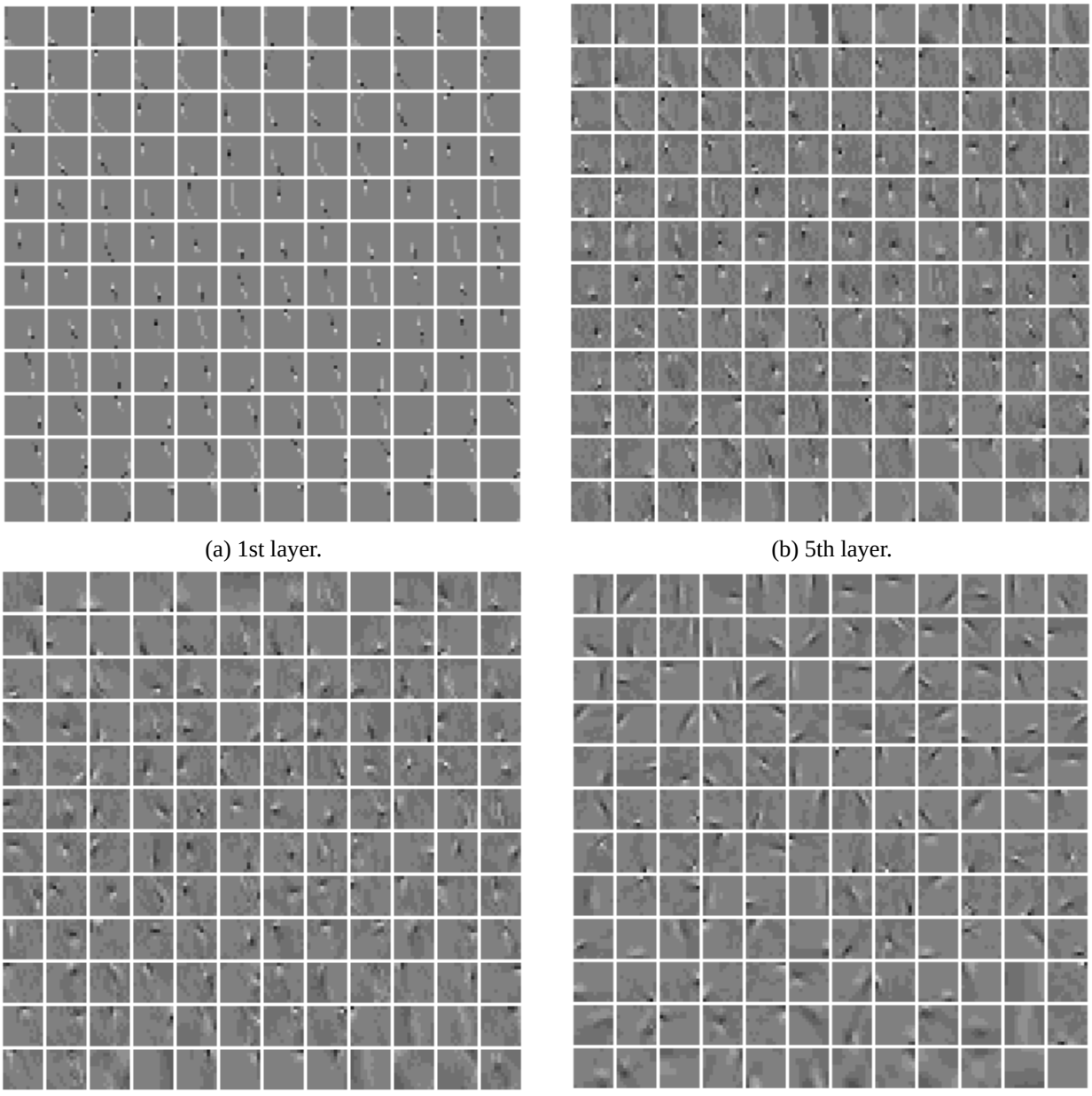
[10] Hyvärinen, A. & Oja, E. (1997). A fast fixed-Point algorithm for independent component analysis. *Neural Computation,* 9, 1483-1492.

[11] Jutten, C. & Herault, J. (1991). Blind separation of sources, Part I: An adaptive algorithm based on neuromimetic architecture. *Signal Processing,* 24, 1-10.

[12] Matsuda, Y., & Yamaguchi, K. (2002). An efficient MDS-based topographic mapping algorithm. preprint.

[13] Szu, H. & Hartley, R. (1987). Fast simulated annealing. *Physics Letters,* 122, 157-162.

[14] van Hateren, J. H. & Ruderman, D. L. (1998). Independent component analysis of natural image sequences yields spatiotemporal filters similar to simple cells in primary visual cortex. *Proceedings of the Royal Society of London ser. B,* 265: 2315-2320.



(a) 1st layer.

(b) 5th layer.

(c) 20th layer.

(d) standard ICA.

Fig. 3. Extraction of edge detectors by the linear multilayer ICA algorithm: 50000 samples of 12×12 image patches of natural scenes were given as the observed signals $\boldsymbol{X}$. That is, N and M was 144 and 50000. In the linear multilayer ICA algorithm, the size of local modules n and the max number of layers was set 9 and 20, respectively. In the local ICA phase, $g(u) = u \exp(-\frac{u^2}{2})$ was used as the nonlinearity. The other settings were the same in Section 3.1.1. (a): It displays the basis vectors generated by the linear multilayer ICA algorithm at the 1st layer. (b): at the 5th layer. (c): at the 20th layer. (d): It shows the ones by the standard fast ICA algorithm with $g(u) = u \exp(-\frac{u^2}{2})$.