Journal of Electronic Imaging 7(1), 86–93 (January 1998).

Maximum *a posteriori* restoration of blurred images using self-organization

Cheng-Yuan Liou
Wen-Pin Tai
National Taiwan University
Department of Computer Science and Information Engineering
Taipei 10764, Taiwan
E-mail: cyliou@csie.ntu.edu.tw

Abstract. *We use the "magic TV" network with the maximum a posteriori (MAP) criterion to restore a space-dependent blurred image. This network provides a unique topological invariance mechanism that facilitates the identification of such space-dependent blur. Instead of using parametric modeling of the underlying blurred image, we use this mechanism to accomplish the restoration. The restoration is reached by a self-organizing evolution in the network, where the weight matrices are adapted to approximate the blur functions. The MAP criterion is used to indicate the goodness of the approximation and to direct the evolution of the network.* [S1017-9909(98)01001-0]

1 Introduction

Restoration of a blurred image can be solved by removing the blurs, which are usually caused by an out-of-focus camera, linear motion, and the atmospheric turbulence, from the observed image. Blur identification methods have been developed to estimate the unknown blur function in the blurring model, where it is defined as the convolution of an original image with a point spread function (PSF) plus an observation noise. Some methods have focused on simple blurs.[1,2] Many restoration methods based on the parametric techniques model the original image as a 2-D autoregressive moving average (ARMA) process and impose certain statistical assumptions on the image.[3] Such methods formulated the blur identification problem into the parameter estimation problem. The results are widely diverse according to those assumptions made about the model and the image. Nonparametric methods[4–6] that employ certain criteria and solve the restoration problem under basic constraints on the PSF have achieved different results.

Several potential methods[7,8] employ the entropy-related criteria to solve the blur identification problem. In Ref. 7, the unknown blur function is considered as a probability density function and is solved under its *a priori* knowledge. Since the PSF serves as a density function, the constraints are made to the PSF, for example, nonnegativity, finite support, and energy preserving. The solution must obey these constraints. In Ref. 8, the probability density of the PSF is also taken into account for blur identification. Maximizing entropy subject to these constraints gives a solution where the PSF tends to satisfy *a priori* given properties.

We study space-dependent blur functions that also obey the preceding basic constraints. We use a self-organizing network[9] following the idea of the "magic TV" and use the maximum *a posteriori* (MAP) criterion to evolve the network toward the solution. The "magic TV" provides a natural mechanism to utilize the invariant hidden topology in the image data. All blurs (candidates) that meet this invariant topology are learned in the network. This criterion guides the network toward the solution.

In the following two sections, we briefly review the image and the blur model. The self-organizing network and the MAP criterion are also introduced. Following the criterion, we derive the training rule for the network to reach the unknown blur functions. Applications of the network are presented in Sec. 4. The SNR improvement will be used to measure the restoration performance. Based on this measure, we make comparisons between the proposed approach and other methods, including inverse filters, Wiener filters, constrained least squares, Kalman filters, and constrained adaptive restoration.

2 Network and the MAP Criterion

We devise a self-organizing network to learn the blur functions. The self-organizing network contains $\mathcal{N}=N\times N$ neurons arranged in a 2-D plane. Each neuron $\mathbf{i}$, $\mathbf{i}\in\mathcal{N}$, has its own weight matrix $\mathbf{W_i}$, $\mathbf{W_i}\in M_{m\times m}$. Each weight matrix corresponds to a possible solution for an unknown blur matrix. For blurred image $\mathbf{x}_s, \mathbf{x}_s\in\mathcal{X}, \mathcal{X}\subset R^m$, the corresponding estimated image data $\hat{\mathbf{y}}_s, \hat{\mathbf{y}}_s\in\mathcal{Y}, \mathcal{Y}\subset R^m$, will be restored by the best-matching weight matrix $\mathbf{W_c}, \mathbf{c}\in\mathcal{N}$, using

$$\hat{\mathbf{y}}_s=\mathbf{W_c}^{-1}\mathbf{x}_s. \tag{1}$$

The components in the vector $\mathbf{x}_s$ are composed of the data in the $m_1\times m_2$ image matrix $\mathbf{X}_s$, $m_1\cdot m_2=m$, from one rectangular region of the blurred image, i.e., $\mathbf{x}_s\equiv\mathrm{vec}(\mathbf{X}_s)$ in lexicographically ordered form,[10] where the $\mathrm{vec}(\mathbf{X}_s)$ transforms the matrix $\mathbf{X}_s$ into a vector by stacking

Paper CIP-10 received Feb. 10, 1997; revised manuscript received July 1, 1997; accepted for publication Aug. 1, 1997.
1017-9909/98/$10.00

the columns of $\mathbf{X}_s$ one underneath the other. Solving the unknown blur matrix $\mathbf{W_c}$ in Eq. (1) is an inverse problem of the observation process

$$\mathbf{x}_s = \mathbf{F}_s \mathbf{y}_s + \mathbf{b}_s , \tag{2}$$

where $\mathbf{y}_s$ is the original image data, $\mathbf{F}_s$ is the real space-dependent blur matrix, and $\mathbf{b}_s$ is the noise vector. In Eq. (2), each row of $\mathbf{F}_s$ corresponds to a PSF. The data $\mathbf{x}_s$ may be contaminated by both the blur and the noise. We attempt to solve a $\hat{\mathbf{y}}_s$, which is close to $\mathbf{y}_s$, $\hat{\mathbf{y}}_s \simeq \mathbf{y}_s$, from Eq. (1).

The object is to solve the best-matching matrix $\mathbf{W_c}$ and to derive the original undegraded image data $\mathbf{y}_s$ from the observed $\mathbf{x}_s$. The difficulty with it is lack of knowledge about both the unknown blur matrix and the original image data.[1] Additional constraints are needed in recovering the image. The basic constraints on the PSF can be accomplished easily by limiting and normalizing each $\mathbf{W_i}$ properly. The probability densities for all the possible candidates $\mathbf{W_i}$, $\mathbf{i} \in \mathcal{N}$, can be estimated by counting the excitation frequencies for all neurons. With these densities and the noise model of Eq. (2), we can construct an MAP criterion.

The network is inspired by the ''magic TV.'' [9] It provides a platform and mechanism for exploring the hidden topology under severe transformation. The topological invariance between the input image and the mapped image is the major feature of this mechanism. We utilize this invariance to assist the restoration. This network is devised as a self-organized mapping system, which could identify blur features from the input, i.e., the blurred image data. According to the ''magic TV,'' a point source that is randomly excited in the input image plane will project a blur feature on the output image plane when the blur aperture is in between the input and the output plane. The implicit topology order of these random point sources can be aligned in a 2-D plane according to their coordinates. The excitations of the corresponding features can also be aligned in a similar topology on the network plane through a self-organizing scheme (see Fig. 1). The noisy parts of the blur functions do not have such hidden topology. They will be screened out by the ''magic TV'' mechanism. Thus, we can regularly array these neurons on a rectangular plane with their weight matrices representing (responding to) the blur features.

To achieve statistic average, we use the MAP criterion as the distance measure instead of the linear distance used in the formal self-organization.[9] The MAP criterion is used to select the best-matching neuron $\mathbf{c}$ and to derive the learning rule accordingly to evolve the weights toward the unknown blur functions. Given a blurred image data $\mathbf{x}_s$, we plan to determine a $\mathbf{W_c}$ that maximizes the *a posteriori* probability. The best-matching neuron $\mathbf{c}$ among all neurons is determined by

$$\mathbf{c} = \arg \max_{\mathbf{i} \in \mathcal{N}} \{ p(\mathbf{W_i} | \mathbf{x}_s) \}. \tag{3}$$

According to the Bayesian analysis, the *a posteriori* probability of $\mathbf{W_i}$ given $\mathbf{x}_s$, $p(\mathbf{W_i}|\mathbf{x}_s)$, can be written in the form

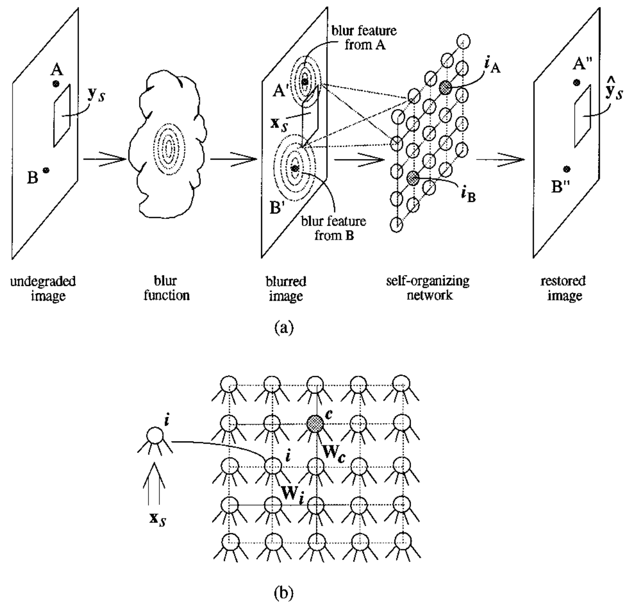


Fig. 1 (a) The concept of the ''magic TV'' for image restoration and (b) the weight matrices in the self-organizing network.

$$p(\mathbf{W_i}|\mathbf{x}_s) = \frac{p(\mathbf{W_i}) p(\mathbf{x}_s|\mathbf{W_i})}{p(\mathbf{x}_s)} \propto p(\mathbf{W_i}) p(\mathbf{x}_s|\mathbf{W_i}). \tag{4}$$

To use this criterion, both the conditional probability $p(\mathbf{x}_s|\mathbf{W_i})$ and the *a priori* probability $p(\mathbf{W_i})$ should be obtained in advance. The probability $p(\mathbf{x}_s)$ is assumed to be a known constant and is omitted from our approach.

To determine the *a priori* probability $p(\mathbf{W_i})$, we count the excitations (number of times of best-matching) for each neuron $\mathbf{i}$, when a batch of input $\{\mathbf{x}_s\}$ are fed to the network. These excitations are accumulated for each neuron. We then normalize these excitation frequencies. The normalized frequency $\hat{R}_\mathbf{i}$ is then used as an estimated $p(\mathbf{W_i})$.

Assume the noise in Eq. (2) is white, zero-mean, and with variance σ_n^2. With a $\hat{\mathbf{y}}_s$ estimated by Eq. (1), the conditional probability $p(\mathbf{x}_s|\mathbf{W_i})$ in Eq. (4) can be formulated as

$$p(\mathbf{x}_s|\mathbf{W_i}) \propto \exp[-E(\mathbf{W_i})], \tag{5}$$

where

$$E(\mathbf{W_i}) = (\mathbf{x}_s - \mathbf{W_i}\hat{\mathbf{y}}_s)^t \mathbf{D} (\mathbf{x}_s - \mathbf{W_i}\hat{\mathbf{y}}_s), \tag{6}$$

and

$$\mathbf{D} = \mathrm{diag}(1/\sigma_n^2, 1/\sigma_n^2, \ldots, 1/\sigma_n^2). \tag{7}$$

3 Synapse Adaptation in the Network

With the MAP criterion, we derive the adaptation rules for the weight matrices in a generalized self-organization sense.[12] First, we determine the best-matching neuron $\mathbf{c}$ for each blurred input data $\mathbf{x}_s$ using the formula

$$
\begin{aligned}
\mathbf{c} &= \arg\max_{\mathbf{i}} \{p(\mathbf{W_i}|\mathbf{x}_s)\} \\
&= \arg\max_{\mathbf{i}} \{p(\mathbf{x}_s|\mathbf{W_i})p(\mathbf{W_i})\} \\
&= \arg\max_{\mathbf{i}} \{\exp[-E(\mathbf{W_i})]\hat{R}_{\mathbf{i}}\} \\
&= \arg\min_{\mathbf{i}} \{E(\mathbf{W_i}) - \log(\hat{R}_{\mathbf{i}})\}. \qquad (8)
\end{aligned}
$$

The best-matching neuron $\mathbf{c}$, which has the minimum value of $E(\mathbf{W_i})-\log(\hat{R}_i)$, will be further used. We then substitute this $\mathbf{W_c}$ in Eq. (1) to estimate a temporary value for $\hat{\mathbf{y}}_s$.

Then we tune the weight matrices to approach the possible blur matrices. The synapse adaptation rules are derived by reducing $E(\mathbf{W_c})-\log(\hat{R}_{\mathbf{c}})$ in Eq. (8) with respect to the tuning of $\mathbf{W_c}$. We apply the steep descent method to update $\mathbf{W_c}$. The weight matrices are adapted by

$$
\begin{aligned}
\mathbf{W}_{\mathbf{c}}^{\text{new}} &= \mathbf{W_c} + \alpha k_{\mathbf{c}} \mathbf{D} \cdot (\mathbf{x}_s - \mathbf{W_c}\hat{\mathbf{y}}_s)\hat{\mathbf{y}}_s^t \\
&= \mathbf{W_c} + \alpha' k_{\mathbf{c}} (\mathbf{x}_s - \mathbf{W_c}\hat{\mathbf{y}}_s)\hat{\mathbf{y}}_s^t, \qquad (9)
\end{aligned}
$$

for the best-matching neuron $\mathbf{c}$, where α and α' are the adaptation rates, and $k_{\mathbf{c}}$ is the scale factor for the adaptation, and by

$$
\begin{aligned}
\mathbf{W}_{\mathbf{i}}^{\text{new}} &= \mathbf{W_i} + \alpha' k_{\mathbf{i}} (\mathbf{x}_s - \mathbf{W_i}\hat{\mathbf{y}}_s)\hat{\mathbf{y}}_s^t \qquad (10) \\
&\approx \mathbf{W_i} + \alpha' k_{\mathbf{i}} (\mathbf{W_c} - \mathbf{W_i})\hat{\mathbf{y}}_s\hat{\mathbf{y}}_s^t \\
&\approx \mathbf{W_i} + \alpha' h_{\mathbf{ci}} (\mathbf{W_c} - W_{\mathbf{i}}), \qquad (11)
\end{aligned}
$$

for all $\mathbf{i} \in \mathscr{H}(\mathbf{c})$, $\mathbf{i} \neq \mathbf{c}$, where $\mathscr{H}(\mathbf{c})$ is the effective region of the neighborhood function $h_{\mathbf{ci}}$ in the self-organization. Since the diagonal component of $\mathbf{D}$ is an unknown constant in Eqs. (9) and (11), $\alpha\mathbf{D}$ can be set to a new adaptation rate α'. This unknown constant will not affect the choice of $\mathbf{c}$ in Eq. (8). From here on we set $\sigma_n^2 = 1$.

The value of k in Eq. (9) can be intuitively set to

$$
k_{\mathbf{c}} \equiv \frac{1}{\sigma_{\mathbf{x}_s}^2}, \qquad (12)
$$

where $\sigma_{\mathbf{x}_s}^2$ is the variance of the data $\mathbf{x}_s$. This means that we update Eq. (9) safely in a smooth region and carefully in a rough region. Experiments show that this is a good assignment.

The value $k_{\mathbf{i}}$ is introduced to scale the adaptation of $\mathbf{W_i}$ in Eq. (11). If the distance between neuron $\mathbf{i}$ and neuron $\mathbf{c}$ is large, the weight matrix $\mathbf{W_i}$ could not be a candidate of the current blur function. The scaling of the weight update is needed for accurate mapping. We can set the neighborhood function $h_{\mathbf{ci}}$ to keep the relationship between $k_{\mathbf{c}}$ and $k_{\mathbf{i}}$ and use it in the adaptation rule of Eq. (11) for $\mathbf{W_i}$, $\mathbf{i} \neq \mathbf{c}$. This neighborhood function is assigned to a general lateral interaction function in formal self-organization. Furthermore, this adaptation rule has less computation load than that of the rule in Eq. (10).

After the adaptation by Eqs. (9) and (11), the weight matrices are modified to satisfy the basic constraints on the PSF. By the self-organizing learning, various blur functions in the rows are obtained for the space-dependent identification.

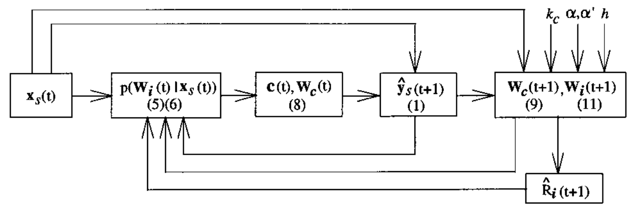


Fig. 2 System chart for self-organization.

In summary, there are six steps in one training iteration:

1. Find the best-matching neuron $\mathbf{c}$ by Eq. (8) for a given input data $\mathbf{x}_s$.
2. Estimate the restored image data $\hat{\mathbf{y}}_s$ from $\mathbf{W_c}$ and $\mathbf{x}_s$ by Eq. (1).
3. Tune the values of the training parameters α and h.
4. Adjust the weight matrix $\mathbf{W_c}$ by Eq. (9).
5. Adjust the weight matrices $\mathbf{W_i}$, $\mathbf{i} \in \mathscr{H}(\mathbf{c})$, $\mathbf{i} \neq \mathbf{c}$, by Eq. (11).
6. Modify the weight matrices with respect to the basic constraints on the PSF, nonnegativity and energy preserving.

Figure 2 shows a flow chart in an iteration.

4 Simulation Results

We now apply this approach to restore the blurred real images in presence or absence of noise. The simulations are carried out for a scanned 64×64 8-bit monochrome image. The original image in absence of noise is shown in Fig. 3. The image in Fig. 3 are sampled from the picture of ''Lena.'' An image degraded by additive white Gaussian noise with 20-dB signal-to-noise ratio (SNR) is shown in Figure 4. The SNR in the noisy image is defined by

Fig. 3 Original 64×64 8-bit image for simulations.

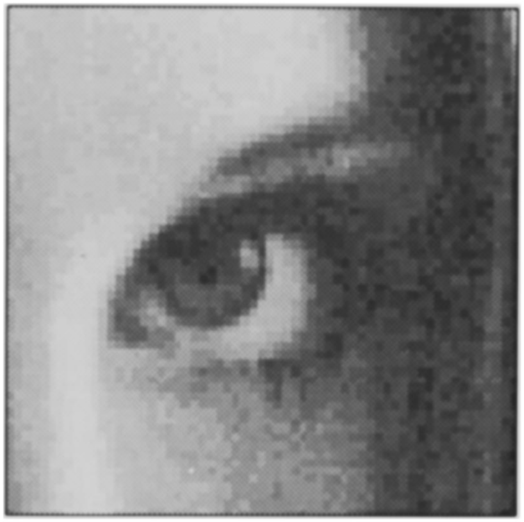


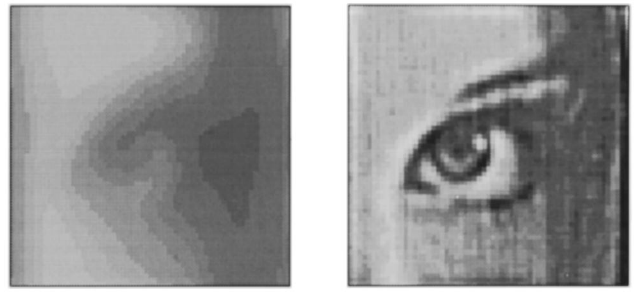
Fig. 4 Image of Fig. 3 degraded by white Gaussian noise at 20 dB SNR.

$$\mathrm{SNR}=10\log_{10}\frac{\sigma_{\mathbf{x}_s}^2}{\sigma_n^2}, \tag{13}$$

where $\sigma_{\mathbf{x}_s}^2$ is the variance of the blurred image $\mathbf{x}_s$ and σ_n^2 is the variance of the white Gaussian noise.

In the simulation network, there are 16×16 neurons with $16\times16\times121\times121$ weights, each neuron has a 121×121-dimensional weight matrix, which resembles the 121×121 blur feature matrix. In this case, $N=16$ and $m_1=m_2=11$. The initial weight matrices are set to variously normalized 2-D Gaussian-shaped functions. The total iterations are set to 500. In the training, the adaptation rate α' is kept constant in 0.1 and the effective region of the neighborhood function h shrinks gradually. All simulations are executed on a personal computer with a 200-MHz processing clock rate.

Without any information about the true PSF, assumptions on the blur function are made. We assume that the support of the blur function $\mathbf{W_i}$ is larger than that of the true PSF. The proper support size for $\mathbf{W_i}$ can be decided from experience.

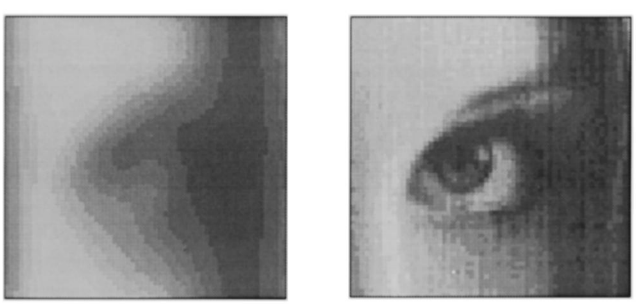


(a) (b)

Fig. 6 (a) Image of Fig. 3 degraded by 7×7 blur functions and (b) restored image for the degraded image in (a). The value of Δ_{SNR} is 3.3094.

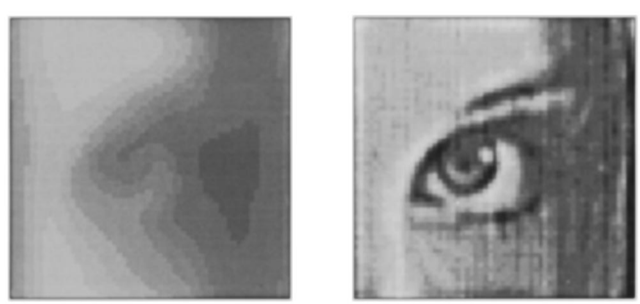
Since the true PSFs are totally unknown, the initial weight matrices $\mathbf{W_i}$ can be set to normalized random values. With such initialization, the network will require more iterations to reach the optimal solution. To ease the convergence, we set the initial weight matrices to various normalized 2-D Gaussian-shaped functions. Before the training, each neuron has a normalized 2-D Gaussian function with random standard deviation (σ_1^2,σ_2^2). When the value $(\sigma_1^2+\sigma_2^2)^{1/2}$ is vary large, the blur function will be a uniform function in the support area. Conversely, when the value is small, the blue function will be similar to an impulse function. All trained results are not significantly affected by such initialization.

The blurred images and the restored images are shown in Figs. 5 to 9. In part (a) of each figure, the blurred image is generated using different blur matrices where each row is assigned to a normalized 2-D ''noisy'' Gaussian function where the noise is set to white and random with a strength of 30-dB SNR. The images from Fig. 3 degraded by 9×9 and 7×7 blur functions are shown in Fig. 5(a) and Fig. 6(a), respectively. The image of Fig. 3 degraded by 7×7 blur functions in presence of noise at 30 dB SNR is shown in Fig. 7(a). The image of Fig. 4 degraded by 7×7 blur functions is shown in Fig. 8(a). The image of Fig. 4 degraded by 7×7 blur functions in presence of noise at 30 dB SNR is shown in Fig. 9(a). Part (b) of each figure shows the restored image. The average computation time costs 1.17 h for each case. The learning curves for the five cases are shown in Figs. 10(a) to 10(e). The averaged values of



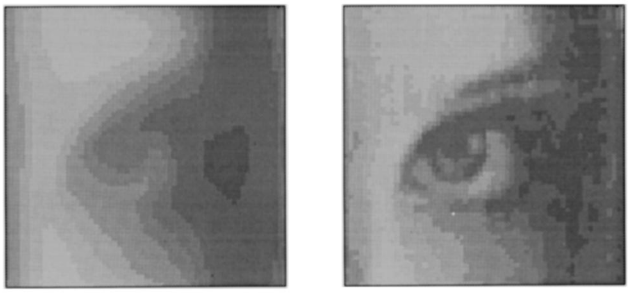
(a) (b)

Fig. 5 (a) Image of Fig. 3 degraded by 9×9 blur functions and (b) restored image for the degraded image in (a). The value of Δ_{SNR} is 9.8068.



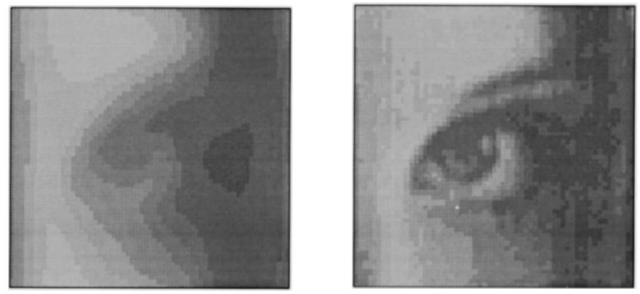
(a) (b)

Fig. 7 (a) Image of Fig. 3 degraded by 7×7 blur functions in presence of white Gaussian noise at 30 dB SNR and (b) restored image for the degraded image in (a). The value of Δ_{SNR} is 3.2535.



(a) (b)

Fig. 8 (a) Image of Fig. 4 degraded by 7×7 blur functions and (b) restored image for the degraded image in (a). The value of Δ_{SNR} is 7.6621.



(a) (b)

Fig. 9 (a) Image of Fig. 4 degraded by 7×7 blur functions in presence of white Gaussian noise at 30 dB SNR and (b) restored image for the degraded image in (a). The value of Δ_{SNR} is 5.7993.

$E(\mathbf{W_c})$ for all $\mathbf{x}_s$ are plotted to display the learning behavior.

The convergence rate is related to the setup of the training parameters, including the total iterations, the adaptation rate, and the neighborhood function. In all our simulations, the weight matrices are always converged and approximate the true blur functions. The learning curves are always declined and stabilized after certain iterations. During each

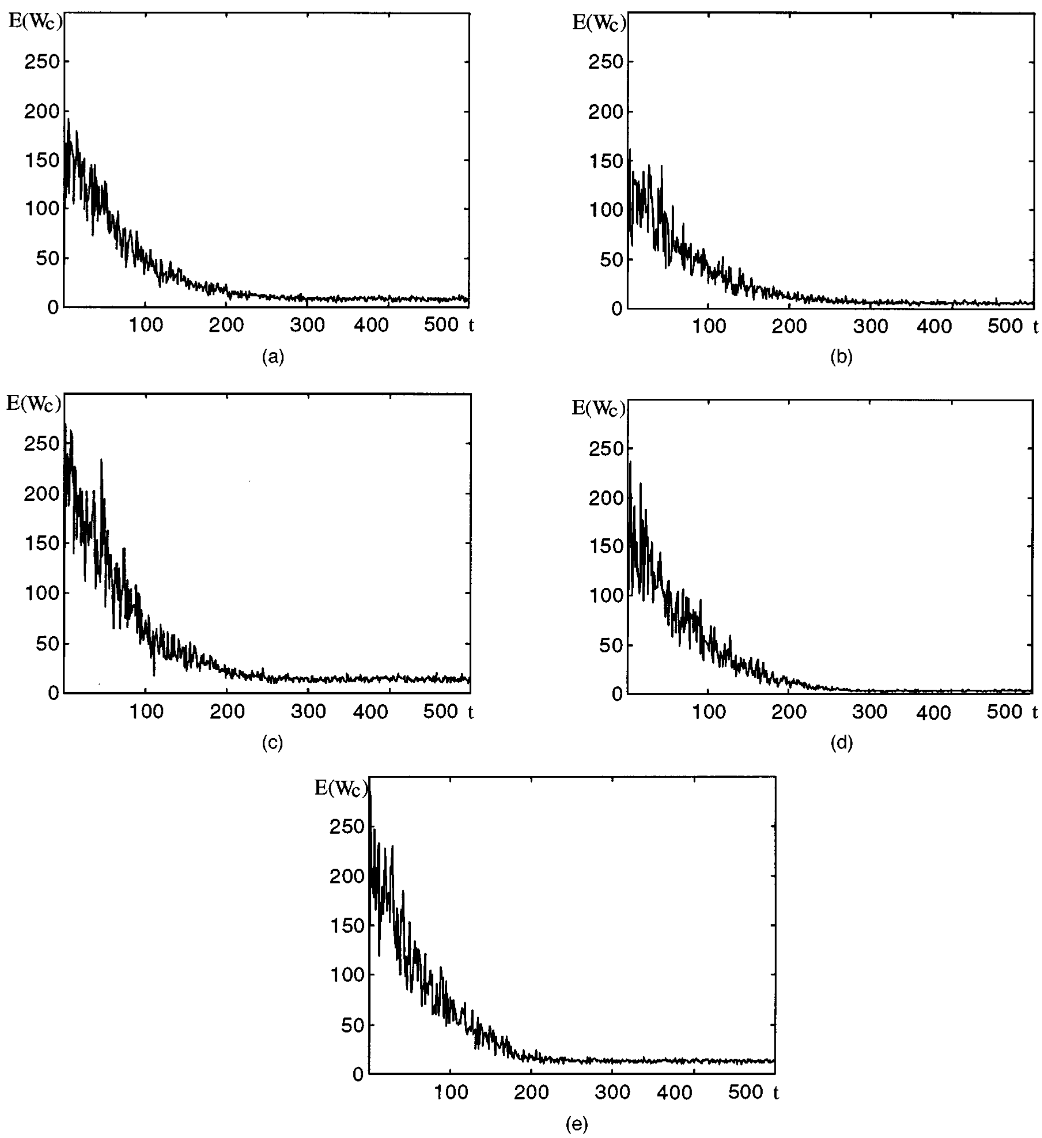


Fig. 10 The learning curves for the five cases in Figs. 5 to 9, respectively.

Fig. 11 Original 256×256 8-bit ''Cameraman'' image.

Fig. 12 Image of Fig. 11 degraded by white Gaussian noise at 30 dB SNR.

(a) (b)

Fig. 13 (a) Image of Fig. 11 degraded by 5×5 blur functions and (b) restored image for the degraded image in (a). The value of Δ_{SNR} is 8.4989.

(a) (b)

Fig. 14 (a) Image of Fig. 11 degraded by 3×3 blur functions and (b) restored image for the degraded image in (a). The value of Δ_{SNR} is 4.0347.

iteration, we calculate the averaged values of $E(\mathbf{W_c})$ over all sample input to indicate the goodness of the training results. The convergence rate is mainly related to the variance of the image data and the initialization of the weight matrices.

SNR improvement is a popular measurement for the restoration performance. SNR improvement is defined as

$$\Delta_{\mathrm{SNR}} = 10 \log_{10} \frac{\Sigma_s \|\mathbf{y}_s - \mathbf{x}_s\|^2}{\Sigma_s \|\mathbf{y}_s - \hat{\mathbf{y}}_s\|^2}, \tag{14}$$

where $\|\mathbf{y}_s - \mathbf{x}_s\|$ measures the distance between the sampled blurred data $\mathbf{x}_s$ and its corresponding original image data $\mathbf{y}_s$, and $\|\mathbf{y}_s - \hat{\mathbf{y}}_s\|$ measures the distance between the restored data $\hat{\mathbf{y}}_s$ and the original image data $\mathbf{y}_s$. In the simulations, SNR improvements for the five cases are 9.8068, 3.3094, 3.2535, 7.6621, and 5.7993, respectively.

To compare with other methods, this approach is tested with the standard 256×256 8-bit monochrome cameraman image (Fig. 11). Figure 12 shows the ''Cameraman'' image degraded by additive white noise at 30 dB SNR. There are 16×16 neurons in the network. Each neuron has a 49 ×49-dimensional weight matrix. In this case, $N=16$ and $m_1 = m_2 = 7$. The blurred images and the restored images are displayed in Figs. 13 to 17. The noise in the 2-D Gaussian blur function is set to white and random with strength 40 dB SNR. The images from Fig. 11 degraded by 5×5 blur functions and 3×3 blur functions are shown in Fig. 13(a) and 14(a), respectively. The image of Fig. 11 degraded by 3×3 blur functions in presence of noise at 40 dB SNR is shown in Fig. 15(a). The image of Fig. 12 degraded by 3×3 blur functions is shown in Fig. 16(a). The image of Fig. 12 degraded by 3×3 blur functions in presence of noise at 40 dB SNR is shown in Fig. 17(a). Part (b) of each figure shows the restored image. The SNR improvements for the five cases are 8.4989, 4.0347, 4.0285, 4.1148, and 4.0756, respectively. The average computation time costs 3.25 h for each case.

Comparisons are also made with the results produced using other existing methods. These methods include inverse filters, Wiener filters, constrained least squares, Kalman filters, and constrained adaptive restoration.[13] These are classical deblurring methods. The types of blur functions are known *a priori* in these methods. The SNR im-

(a) (b)

Fig. 15 (a) Image of Fig. 11 degraded by 3×3 blur functions in presence of white Gaussian noise at 40 dB SNR and (b) restored image for the degraded image in (a). The value of Δ_{SNR} is 4.0285.

(a) (b)

Fig. 16 (a) Image of Fig. 12 degraded by 3×3 blur functions and (b) restored image for the degraded image in (a). The value of Δ_{SNR} is 4.1148.

(a) (b)

Fig. 17 (a) Image of Fig. 12 degraded by 3×3 blur functions in presence of white Gaussian noise at 40 dB SNR and (b) restored image for the degraded image in (a). The value of Δ_{SNR} is 4.0756.

Table 1 Restoration performance for different methods.[13]

Methods	Δ_{SNR}
Inverse Filters	−16.5
Wiener Filters	5.9
Constrained Least Squares	6.2
Kalman filters	5.6
Constrained Adaptive Restoration	8.1
Our Approach	4.95

provements of the blurred cameraman image for each method are listed in Table 1 for comparisons.

From the simulation results, this approach is robust to the noise in either the observation process, the blur function, or the image data. Comparing the restored results in Fig. 6(b) with those in Figs. 7(b) and 8(b), we find that the noise in the observation process (at 30 dB SNR) causes more difficulty in restoration than the noise in the image data (at 20 dB SNR). Furthermore, this approach has larger SNR improvements for the cases without the observation noise.

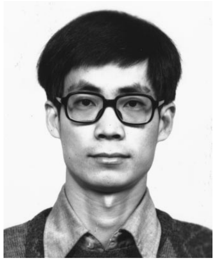
In deriving the learning rule, we attempt to reduce the observation noise. The best-matching neuron **c** for input $\mathbf{x}_s$ is the neuron that has the minimum $E(\mathbf{W_i})-\log(\hat{R}_\mathbf{i})$ value. This $E(\mathbf{W_i})$ is the distance between $\mathbf{x}_s$ and $\mathbf{W_i}\hat{\mathbf{y}}_s$. Minimizing this $E(\mathbf{W_i})$ will reduce the observation noise optimally. The neuron with the minimum $E(\mathbf{W_i})$ value may not win the competition. Hence the learning rule is not the optimal rule for reducing the observation noise. The value $\hat{R}_\mathbf{i}$, which is used to estimate the *a priori* probability $p(\mathbf{W_i})$, is introduced in the term $E(\mathbf{W_i})-\log(\hat{R}_\mathbf{i})$ by Eq. (4). This $\hat{R}_\mathbf{i}$ makes the degraded image with the observation noise difficult to restored. We may prefer the neurons with equal probabilities to improve the performance.

Acknowledgment

This work has been supported by the National Science Council, ROC.

References

1. T. G. Stockham, Jr., T. M. Cannon, and R. B. Ingebretsen, ''Blind deconvolution through digital signal processing,'' *Proc. IEEE* **63**(4), 678–692 (1975).
2. H. C. Andrews, ''Digital image restoration: a survey,'' *Computer* **7**(5), 36–45 (1974).
3. A. M. Tekalp, H. Kaufman, and J. Woods, ''Identification of image and blur parameters for the noncausal blurs,'' *IEEE Trans. Acoust. Speech Signal Process.* **34**, 963–972 (1986).
4. G. R. Ayers and J. C. Dainty, ''Iterative blind deconvolution method and its application,'' *Opt. Lett.* **13**(7), 547–549 (1988).
5. B. C. McCallum, ''Blind deconvolution by simulated annealing,'' *Opt. Commun.* **75**(2), 101–105 (1990).
6. D. Kundur and D. Hatzinakos, ''A novel recursive filtering method for blind image restoration,'' in *Proc. IASTED Int. Conf. on Signal and Image Processing*, pp. 428–431 (1995).
7. R. A. Wiggins, ''Minimum entropy deconvolution,'' *Geoexploration* **16**, 21–35 (1978).
8. M. K. Nguyen and A. Monhammad-Djafari, ''Bayesian approach with the maximum entropy principle in image reconstruction from microwave scattered field data,'' *IEEE Trans. Pattern Anal. Mach. Intell.* **6**, 721–741 (1984).
9. T. Kohonen, *Self-Organization and Associative Memory*, Springer-Verlag, Berlin (1984).
10. H. C. Andrews and B. R. Hunt, *Digital Image Restoration*, Prentice-Hall, Englewood Cliffs, NJ (1977).
11. S. Geman and D. Geman, ''Stochastic relaxation, Gibbs distributions and the Bayesian restoration of images,'' *IEEE Trans. Pattern Anal. Mach. Intell.* **6**, 721–741 (1984).
12. T. Kohonen, ''Generalizations of the self-organizing maps,'' in *Proc. Int. Joint Conf. on Neural Networks*, pp. 457–462, Japan (1993).
13. J. Biemond, L. Lagendijk, and R. M. Mersereau, ''Iterative methods for image deblurring,'' *Proc. IEEE* **78**(5), 856–883 (1990).
14. D. Kundur and D. Hatzinakos, ''Blind image deconvolution,'' *IEEE Signal Process. Mag.* **13**(3), 43–64 (1996).



Cheng-Yuan Liou received the BS degree in physics from National Central University in 1974, the MS degree in physical oceanography from National Taiwan University in 1978, and the PhD degree in ocean engineering from the Massachusetts Institute of Technology in 1985. From 1986 to 1987, he was a visiting associate professor in the Institute of Applied Mechanics, National Taiwan University, where he taught courses in stochastic process and digital signal processing and did research in sonar array signal processing. In 1988, he joined the faculty of the same university, where he is currently a professor in the Department of Computer Science and Information Engineering. His current interests include neural networks, digital signal processing, and artificial life.

Wen-Pin Tai received the BS degree, MS degree and PhD degree in computer science and information engineering from National Taiwan University in 1990, 1992, and 1997, respectively.