USC-SIPI REPORT #112

Image Restoration Using a Neural Network

by

Y.T. Zhou, R. Chellappa and B.K. Jenkins

Signal and Image Processing Institute
UNIVERSITY OF SOUTHERN CALIFORNIA
Department of Electrical Engineering-Systems
Powell Hall of Engineering
University Park/MC-0272
Los Angeles, CA 90089 U.S.A.

This work is partially supported by the AFOSR Contract No. F49620-87-C-007.

Image Restoration Using a Neural Network[1]

Y. T. Zhou, R. Chellappa and B. K. Jenkins

Signal and Image Processing Institute

Department of EE-Systems

University of Southern California

Abstract

A new approach for restoration of gray level images degraded by a known shift invariant blur function and additive noise is presented using a neural computational network. A neural network model is employed to represent a possibly nonstationary image whose gray level function is the simple sum of the neuron state variables. The restoration procedure consists of two stages: estimation of the parameters of the neural network model and reconstruction of images. During the first stage, the parameters are estimated by comparing the energy function of the neural network with a constrained error function. The nonlinear restoration method is then carried out iteratively in the second stage by using a dynamic algorithm to minimize the energy function of an appropriate neural network. Owing to the model's fault–tolerant nature and computation capability, a high quality image is obtained using this approach. A practical algorithm with reduced computational complexity is also presented. Several computer simulation examples involving

[1]This research work is partially supported by the AFOSR Contract No. F-49620-87-C-0007.

synthetic and real images are given to illustrate the usefulness of our method. The choice of the boundary values to reduce the ringing effect is discussed and comparisons with other restoration methods such as the SVD pseudoinverse filter, minimum mean square error (MMSE) filter and modified MMSE filter using Gaussian Markov random field model are given.

1 Introduction

Image restoration is an important problem in early vision processing to recover an ideal high quality image from a degraded recording. Restoration techniques are applied to remove (1) system degradations such as blur due to optical system aberrations, atmospheric turbulence, motion and diffraction; and (2) statistical degradations due to noise. Over the last 20 years, various methods such as the inverse filter, Wiener filter, Kalman filter, SVD pseudoinverse and many other model based approaches, have been proposed for image restoration. One of the major drawbacks of most of the image restoration algorithms is the computational complexity, so much so that many simplifing assumptions such as wide sense stationarity (WSS), availability of second order image statistics have been made to obtain computationally feasible algorithms. The inverse filter method works only for extremely high signal to noise ratio images. The Wiener filter is usually implemented only after wide sense stationary assumption has been made for images. Furthermore, knowledge of power spectrum or correlation matrix of the undegraded image is

required. Oftentimes, additional assumptions regarding boundary conditions are made so that fast orthogonal transforms can be used. The Kalman filter approach can be applied to nonstationary image but is computationally very intensive. Similar statements can be made for SVD pseudoinverse filter method. Approaches based on noncausal models such as the noncausal autoregressive or Gauss Markov random field models also make assumptions such as WSS and periodic boundary conditions. It is desirable to develop a restoration algorithm that does not make WSS assumptions and can be implemented in a reasonable time. An artificial neural network system that can perform extremely rapid computations seems to be very attractive for image restoration in particular and image processing and pattern recognition [1] in general.

In this paper, we use a neural network model containing redundant neurons to restore gray level images degraded by a known shift invariant blur function and noise. It is based on the model described in [2] [3] using a simple sum number representation [4]. The image gray levels are represented by the simple sum of the neuron state variables which take binary values of 1 or 0. The observed image is degraded by a shift–invariant function and noise. The restoration procedure consists of two stages: estimation of the parameters of the neural network model and reconstruction of images. During the first stage, the parameters are estimated by comparing the energy function of the neural network with the constrained error function. The nonlinear restoration algorithm is then implemented using a dynamic iterative algorithm to minimize the energy function of the

neural network. Owing to the model's fault–tolerant nature and computation capability, a high quality image is obtained using this approach. We illustrate the usefulness of this approach by using both synthetic and real images degraded by a known shift–invariant blur function with or without noise. We also discuss the problem of choosing boundary values and introduce two methods to reduce the ringing effect. Comparisons with other restoration methods such as the SVD pseudoinverse filter, minimum mean square error (MMSE) filter and modified MMSE filter using Gaussian Markov random field model are given using real images. The advantages of the method developed in this paper are (1) WSS assumption is not required for the images, (2) can be implemented rapidly and (3) is fault tolerant.

The organization of this paper is as follows. A network model containing redundant neurons for image reprsentation and the image degradation model are given in Section 2. A technique for parameter estimation is presented in Section 3. Image generation using a dynamic algorithm is described in Section 4. A practical algorithm with reduced computational complexity is presented in Section 5. Computer simulation results using neural network to restore degraded images are given in Section 6. Choice of the boundary values is discussed in Section 7. Comparisons with other methods are given in Section 8 and conclusions and remarks are included in Section 9.

2 A Neural Network for Image Representation

We use a neural network containing redundant neurons for representing the image gray levels. The model consists of $L^2 \times M$ mutually interconnected neurons, where L is the size of image and M is the maximum value of the gray level function. Let $V = \{v_{i,k},\ where\ 1 \leq i \leq L^2, 1 \leq k \leq M\}$ be a binary state set of the neural network with $v_{i,k}$ (1 for firing and 0 for resting) denoting the state of the (i,k)th neuron. Let $T_{i,k;j,l}$ denote the strength (possibly negative) of the interconnection between neuron (i,k) and neuron (j,l). We require symmetry

$$T_{i,k;j,l} = T_{j,l;i,k} \quad for \quad 1 \leq i,j \leq L^2 \ and \ 1 \leq l,k \leq M$$

We also allow for neurons to have self-feedback, i.e. $T_{i,k;i,k} \neq 0$. In this model, each neuron (i,k) randomly and asynchronously receives inputs $\sum T_{i,k;j,l} v_{j,l}$ from all neurons and a bias input $I_{i,k}$

$$u_{i,k} = \sum_{j}^{L^2} \sum_{l}^{M} T_{i,k;j,l} v_{j,l} + I_{i,k} \tag{1}$$

Each $u_{i,k}$ is fed back to corresponding neurons after thresholding

$$v_{i,k} = g(u_{i,k}) \tag{2}$$

where $g(x)$ is a nonlinear function whose form can be taken as

$$g(x) = \begin{cases} 1 & if \ \ x \geq 0 \\ 0 & if \ \ x < 0. \end{cases} \tag{3}$$

In this model, the state of each neuron is updated by using the latest information about other neurons.

The image is described by a finite set of gray level functions $\{x(i,j), \ where \ 1 \leq i,j \leq L\}$ with $x(i,j)$ (positive integer number) denoting the gray level of the pixel (i,j). The image gray level function can be represented by a simple sum of the neuron state variables as

$$x(i,j) = \sum_{k=1}^{M} v_{m,k} \tag{4}$$

where $m = i \times L + j$. Here the gray level functions have degenerate representations. Use of this redundant number representation scheme yields advantages such as fault–tolerance and convergence to the solution [4].

By using the lexicographic notation, the image degradation model can be written as

$$\underline{Y} = H\underline{X} + \underline{N} \tag{5}$$

where

$$H = \begin{bmatrix} h_{1,1} & h_{1,2} & \cdot & \cdot & \cdot & h_{1,L^2} \\ h_{2,1} & h_{2,2} & \cdot & \cdot & \cdot & h_{2,L^2} \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ h_{L^2,1} & h_{L^2,2} & \cdot & \cdot & \cdot & h_{L^2,L^2} \end{bmatrix} \tag{6}$$

is the "blurring matrix" corresponding to a blur function,

$$\underline{N} = \begin{bmatrix} \underline{N}_1 \\ \underline{N}_2 \\ \cdot \\ \cdot \\ \cdot \\ \underline{N}_L \end{bmatrix} = \begin{bmatrix} n_1 \\ n_2 \\ \cdot \\ \cdot \\ \cdot \\ n_{L^2} \end{bmatrix}, \quad \underline{N}_i = \begin{bmatrix} n(i,1) \\ n(i,2) \\ \cdot \\ \cdot \\ \cdot \\ n(i,L) \end{bmatrix} = \begin{bmatrix} n_{(i-1)\times L+1} \\ n_{(i-1)\times L+2} \\ \cdot \\ \cdot \\ \cdot \\ n_{i\times L} \end{bmatrix} \tag{7}$$

is the signal independent white noise,

$$\underline{X} = \begin{bmatrix} \underline{X}_1 \\ \underline{X}_2 \\ \cdot \\ \cdot \\ \cdot \\ \underline{X}_L \end{bmatrix} = \begin{bmatrix} x_1 \\ x_2 \\ \cdot \\ \cdot \\ \cdot \\ x_{L^2} \end{bmatrix}, \quad \underline{X}_i = \begin{bmatrix} x(i,1) \\ x(i,2) \\ \cdot \\ \cdot \\ \cdot \\ x(i,L) \end{bmatrix} = \begin{bmatrix} x_{(i-1)\times L+1} \\ x_{(i-1)\times L+2} \\ \cdot \\ \cdot \\ \cdot \\ x_{i\times L} \end{bmatrix} \tag{8}$$

is the original image and

$$\underline{Y} = \begin{bmatrix} \underline{Y}_1 \\ \underline{Y}_2 \\ \cdot \\ \cdot \\ \cdot \\ \underline{Y}_L \end{bmatrix} = \begin{bmatrix} y_1 \\ y_2 \\ \cdot \\ \cdot \\ \cdot \\ y_{L^2} \end{bmatrix}, \quad \underline{Y}_i = \begin{bmatrix} y(i,1) \\ y(i,2) \\ \cdot \\ \cdot \\ \cdot \\ y(i,L) \end{bmatrix} = \begin{bmatrix} y_{(i-1)\times L+1} \\ y_{(i-1)\times L+2} \\ \cdot \\ \cdot \\ \cdot \\ y_{i\times L} \end{bmatrix} \tag{9}$$

is the degraded image. This is similar to the simultaneous equations solution of [4], but differs in that (5) includes a noise term.

The shift–invariant blur function can be written as a convolution over a small window, for instance, it takes the form

$$h(k,l) = \begin{cases} \frac{1}{2} & if \;\; k = 0,\; l = 0 \\ \frac{1}{16} & if \;\; |k|,\; |l| \leq 1,\;\; (k,l) \neq (0,0) \end{cases} \tag{10}$$

accordingly, the "blur matrix" H will be a block Toeplitz or block circulant matrix (if the image has periodic boundaries). The block circulant matrix corresponding to (10) can be written as

$$H = \begin{bmatrix} H_0 & H_1 & \underline{0} & \cdot & \cdot & \cdot & \underline{0} & H_1 \\ H_1 & H_0 & H_1 & \cdot & \cdot & \cdot & \underline{0} & \underline{0} \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ H_1 & \underline{0} & \underline{0} & \cdot & \cdot & \cdot & H_1 & H_0 \end{bmatrix} \tag{11}$$

where

$$H_0 = \begin{bmatrix} \frac{1}{2} & \frac{1}{16} & 0 & \cdots & 0 & \frac{1}{16} \\ \frac{1}{16} & \frac{1}{2} & \frac{1}{16} & \cdots & 0 & 0 \\ \cdot & \cdot & \cdot & \cdots & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdots & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdots & \cdot & \cdot \\ \frac{1}{16} & 0 & 0 & \cdots & \frac{1}{16} & \frac{1}{2} \end{bmatrix}, \quad H_1 = \begin{bmatrix} \frac{1}{16} & \frac{1}{16} & 0 & \cdots & 0 & \frac{1}{16} \\ \frac{1}{16} & \frac{1}{16} & \frac{1}{16} & \cdots & 0 & 0 \\ \cdot & \cdot & \cdot & \cdots & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdots & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdots & \cdot & \cdot \\ \frac{1}{16} & 0 & 0 & \cdots & \frac{1}{16} & \frac{1}{16} \end{bmatrix} \tag{12}$$

and $\underline{0}$ is null matrix whoes elements are all zeros.

3 Estimation of Model Parameters

The neural model parameters, the interconnection strengths and the bias inputs, can be determined in terms of the energy function of the neural network. As defined in [2], the energy function of the neural network can be written as

$$E = -\frac{1}{2}\sum_{i=1}^{L^2}\sum_{j=1}^{L^2}\sum_{k=1}^{M}\sum_{l=1}^{M} T_{i,k;j,l}\, v_{i,k}\, v_{j,l} - \sum_{i=1}^{L^2}\sum_{k=1}^{M} I_{i,k}\, v_{i,k} \tag{13}$$

In order to use the spontaneous energy–minimization process of the neural network, we reformulate the restoration problem as one of minimizing an error function with constraints defined as

$$E = \frac{1}{2}\|\underline{Y} - H\underline{\hat{X}}\|^2 + \frac{1}{2}\lambda\|D\underline{\hat{X}}\|^2 \tag{14}$$

where $\|\underline{Z}\|$ is the L_2 norm of $\underline{Z}$ and λ is a constant. The first term in (14) is to seek an $\underline{\hat{X}}$ such that $H\underline{\hat{X}}$ approximates $\underline{Y}$ in a least squares sense. Meanwhile, the second term

is a smoothness constraint on the solution $\underline{\hat{X}}$. The constant λ determines their relative importance to achieve both noise suppression and ringing reduction. In general, if H is a low pass distortion, then D is a high pass filter. A common choice of D is a second order differential operator which can be approximated as a local window operator in the 2–D discrete case. For instance, if D is a Laplacian operator

$$\nabla = \frac{\partial^2}{\partial i^2} + \frac{\partial^2}{\partial j^2} \tag{15}$$

it can be approximated as a window operator

$$\frac{1}{6}\begin{bmatrix} 1 & 4 & 1 \\ 4 & -20 & 4 \\ 1 & 4 & 1 \end{bmatrix}. \tag{16}$$

Then D will be a block Toeplitz matrix similar to (11).

By comparing the terms in the expansion of (14) with the corresponding terms in (13), we can determine the interconnection strengths and the bias inputs as

$$T_{i,k;j,l} = -\sum_{p=1}^{L^2} h_{p,i}\, h_{p,j} - \lambda \sum_{p=1}^{L^2} d_{p,i}\, d_{p,j} \tag{17}$$

and

$$I_{i,k} = \sum_{p=1}^{L^2} y_p\, h_{p,i}. \tag{18}$$

where $h_{i,j}$ and $d_{i,j}$ are the elements of the matrices H and D, respectively.

From (17), one can see that the interconnection strengths are determined by the shift–invariant blur function, differential operator and constant λ. Hence, $T_{i,k;j,l}$ can

be computed without error provided the blur function is known. However, the bias inputs are functions of the observed degraded image. If the image is degraded by a shift–invariant blur function only, then $I_{i,k}$ can be estimated perfectly. Otherwise, $I_{i,k}$ is affected by noise. The reasoning behind this statement is as follows. By replacing y_p by $\sum_{i=1}^{L^2} h_{p,i}\, x_i + n_p$, we have

$$\begin{aligned} I_{i,k} &= \sum_{p=1}^{L^2} (\sum_{i=1}^{L^2} h_{p,i}\, x_i + n_p)\, h_{p,i} \\ &= \sum_{p=1}^{L^2} \sum_{i=1}^{L^2} h_{p,i}\, x_i\, h_p + \sum_{p=1}^{L^2} n_p\, h_{p,i} \end{aligned} \tag{19}$$

The second term in (19) represents the effects of noise. If the signal to noise ratio (SNR), defined by

$$SNR = 10\, \log_{10}\, \frac{\sigma_s^2}{\sigma_n^2}, \tag{20}$$

where σ_s^2 and σ_n^2 are variances of signal and noise, respectively, is low, then we have to choose a large λ to suppress effects due to noise.

4 Restoration

Restoration is carried out by neuron evaluation and an image construction procedure. Once the parameters $T_{i,k;j,l}$ and $I_{i,k}$ are obtained using (17) and (18), each neuron can randomly and asynchronously evaluate its state and readjust accordingly using (1) and (2). When one quasi–minimum energy point is reached, the image can be constructed using (4).

However, this neural network has self–feedback, i.e. $T_{i,k;i,k} \neq 0$. As a result the energy function E does not always decrease monotonically with a transition. This is explained below. Define the state change $\Delta v_{i,k}$ of neuron (i,k) and energy change ΔE as

$$\Delta v_{i,k} = v_{i,k}^{new} - v_{i,k}^{old} \quad and \quad \Delta E = E^{new} - E^{old}$$

Consider the energy function

$$E = -\frac{1}{2} \sum_{i=1}^{L^2} \sum_{j=1}^{L^2} \sum_{k=1}^{M} \sum_{l=1}^{M} T_{i,k;j,l}\, v_{i,k}\, v_{j,l} - \sum_{i=1}^{L^2} \sum_{k=1}^{M} I_{i,k}\, v_{i,k}, \tag{21}$$

Then the change ΔE due to a change $\Delta v_{i,k}$ is given by

$$\Delta E = -(\sum_{j=1}^{L^2} \sum_{l=1}^{M} T_{i,k;j,l}\, v_{j,l} + I_{i,k}\,)\, \Delta v_{i,k} - \frac{1}{2}\, T_{i,k;i,k}\, (\Delta v_{i,k})^2 \tag{22}$$

which is not always negative. For instance, if

$$v_{i,k}^{old} = 0, \qquad u_{i,k} = \sum_{j=1}^{L^2} \sum_{l=1}^{M} T_{i,k;j,l}\, v_{j,l} + I_{i,k} \; > 0,$$

and the threshold function is as in (3), then $v_{i,k}^{new} = 1$ and $\Delta v_{i,k} > 0$. Thus, the first term in (22) is negative. But

$$T_{i,k;i,k} = -\sum_{p=1}^{L^2} h_{p,i}^2 - \lambda \sum_{p=1}^{L^2} d_{p,i}^2 < 0.$$

with $\lambda > 0$ leading to

$$-\frac{1}{2}\, T_{i,k;i,k}\, (\Delta v_{i,k})^2 > 0.$$

When the first term is less than the second term in (22), then $\Delta E > 0$ (we have observed this in our experiments), which means E is not a Lyapunov function. Consequently, the convergence of the network is not guaranteed [5].

Thus, depending on whether convergence to a local minimum or a global minimum is desired, we can design a deterministic or stochastic decision rule. The deterministic rule is to take a new state $v_{i,k}^{new}$ of neuron (i,k) if the energy change ΔE due to state change $\Delta v_{i,k}$ is less than zero. If ΔE due to state change is > 0, no state change is affected. We have also designed a stochastic rule similar to the one used in simulated annealing techniques [6] [7]. The details of this stochastic scheme are given as follows:

Define a Boltzmann distribution by

$$\frac{p_{new}}{p_{old}} = e^{\frac{-\Delta E}{T}}$$

where p_{new} and p_{old} are the probabilities of the new and old global state, respectively, ΔE is the energy change and T is the parameter which acts like temperature. A new state $v_{i,k}^{new}$ is taken if

$$\frac{p_{new}}{p_{old}} > 1, \quad or \ if \ \ \frac{p_{new}}{p_{old}} \leq 1 \ \ but \ \ \frac{p_{new}}{p_{old}} > \xi$$

where ξ is a random number uniformly distributed in the interval [0,1].

The restoration algorithm is summarized as below.

Algorithm 1:

1. Set the initial state of the neurons.

2. Update the state of all neurons randomly and asynchronously according to the decision rule.

3. Check the energy function; if energy does not change, go to step 4; otherwise, go back to step 2.

4. Construct an image using (4).

5 A Practical Algorithm

The algorithm described above is difficult to simulate on a conventional computer owing to high computational complexity even for images of reasonable size. For instance, if we have an $L \times L$ image with M gray levels, then L^2M neurons and $\frac{1}{2}L^4M^2$ interconnections are required and L^4M^2 additions and multiplications are needed at each iteration. Therefore, the space and time complexities are $O(L^4M^2)$ and $O(L^4M^2K)$, respectively, where K, $O(10)$–$O(100)$, is the number of iterations. When $L = 256$ and $M = 256$, the space and time complexities will be $O(10^{14})$ and $O(10^{15})$–$O(10^{15})$, respectively. However, simplification is possible if the neurons are sequentially updated .

In order to simplify the algorithm, we begin by reconsidering (1) and (2) of the neural network. Noting that the interconnection strengths given in (17) are independent of subscripts k and l and the bias inputs given in (18) are independent of subscript k, the M neurons used to represent the same image gray level function have the same interconnection strengths and bias inputs. Hence, one set of interconnection strengths and one bias input are sufficient for every gray level function, i.e. the dimensions of the

interconnection matrix T and bias input matrix I can be reduced by a factor of M^2. From (1) all inputs received by a neuron, say, the (i,k)th neuron can be written as

$$\begin{aligned} u_{i,k} &= \sum_{j}^{L^2} T_{i,.;j,.} \, (\sum_{l}^{M} v_{j,l}) + I_{i,.} \\ &= \sum_{j}^{L^2} T_{i,.;j,.} \, x_j + I_{i,.} \end{aligned} \tag{23}$$

where we have used (4) and x_j is the gray level function of the jth image pixel. The symbol "." in the subscripts means that the $T_{i,.;j,.}$ and $I_{i,.}$ are independ of k. Equation (23) suggests that we can use a multivalue number to replace the simple sum number. Since the interconnection strengths are determined by the blur function, differential operator and constant λ as shown in (17), it is easy to see that if the blur function is local, then most interconnection strengths are zeros and the neurons are locally connected. Therefore, most elements of the interconnection matrix T are zeros. If the blur function is shift invariant taking the form in (10), then the interconnection matrix is block Toeplitz so that only a few elements need to be stored. Based on the value of inputs $u_{i,k}$, the state of the (i,k)th neuron is updated by applying a decision rule. The state change of the (i,k)th neuron in turn causes the gray level function x_i to change

$$x_i^{new} = \begin{cases} x_i^{old} & if \;\; \Delta v_{i,k} = 0 \\ x_i^{old} + 1 & if \;\; \Delta v_{i,k} = 1 \\ x_i^{old} - 1 & if \;\; \Delta v_{i,k} = -1 \end{cases} \tag{24}$$

where $\Delta v_{i,k} = v_{i,k}^{new} - v_{i,k}^{old}$ is the state change of the (i,k)th neuron. The supscripts "new" and "old" are for after and before updating, respectively. We use x_i to respresent the

gray level value as well as the output of M neurons representing x_i. Assuming that the neurons of the network are sequentially visited, it is straightforward to show that the updating procedure can be reformulated as

$$u_{i,k} = \sum_{j}^{L^2} T_{i,.;j,.}\, x_j + I_{i,.} \tag{25}$$

$$\Delta v_{i,k} = g(u_{i,k}) = \begin{cases} \Delta v_{i,k} = 0 & if \;\; u_{i,k} = 0 \\ \Delta v_{i,k} = 1 & if \;\; u_{i,k} > 0 \\ \Delta v_{i,k} = -1 & if \;\; u_{i,k} < 0 \end{cases} \tag{26}$$

$$x_i^{new} = \begin{cases} x_i^{old} + \Delta v_{i,k} & if \;\; \Delta E < 0 \\ x_i^{old} & if \;\; \Delta E \geq 0 \end{cases} \tag{27}$$

Note that the stochastic decision rule can also be used in (27). In order to limit the gray level function to the range 0–255 after each updating step, we have to check the value of the gray level function x_i^{new}. Equations (25), (26) and (27) give a much simpler algorithm. This algorithm is summarized below.

Algorithm 2:

1. Take the degraded image as the initial value.

2. Sequentially visit all numbers (image pixels). For each number, use (25), (26) and (27) to update it repeatedly until no further change, i.e. if $\Delta v_{i,k} = 0$ or energy change $\Delta E \geq 0$, then move to next one.

3. Check the energy function; if energy does not change anymore, a restored image is obtained; otherwise, go back to step 2 for another iteration.

The calculations of the inputs $u_{i,k}$ of the (i,k)th neuron and the energy change ΔE can be simplified furthermore. When we update the same image gray level function repeatedly, the input received by the current neuron (i,k) can be computed by making use of the previous result

$$u_{i,k} = u_{i,k-1} + \Delta v_{i,k}\, T_{i,.;i,.} \tag{28}$$

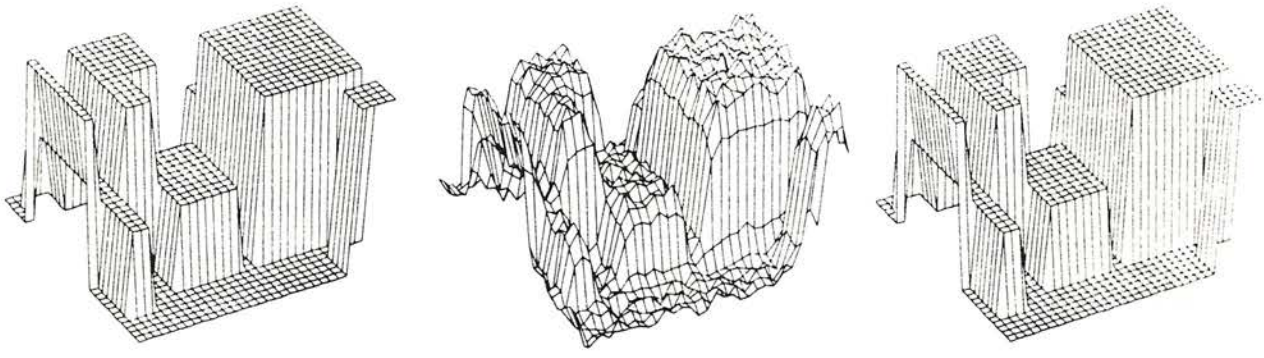
where $u_{i,k-1}$ is the inputs received by the $(i,k-1)$th neuron. The energy change ΔE due to the state change of the (i,k)th neuron can be calculated as

$$\Delta E = -u_{i,k}\, \Delta v_{i,k} - \frac{1}{2}\, T_{i,.;i,.}\, (\Delta v_{i,k})^2 \tag{29}$$

If the blur function is shift invariant, all these simplifications reduce the space and time complexities significantly from $O(L^4M^2)$ and $O(L^4M^2K)$ to $O(L^2)$ and $O(ML^2K)$, respectively. Since every gray level function needs only a few updating steps after the first iteration, the computation at each iteration is $O(L^2)$. The resulting algorithm can be easily simulated on mini–computers for images as large as 512×512.

6 Computer Simulations

The practical algorithm described in the previous section was applied to synthetic and real images on a Sun-3/160 Workstation. In all cases only the deterministic decision rule



(a) Original image. (b) Degraded image. (c) Results after 6 iterations.

Figure 1: Restoration of noisy blurred synthetic image.

was used. The results are summarized in Figure 1 and 2.

Figure 1 shows the results for a synthetic image. The original image shown in Figure 1(a) is of size 32×32 with 3 gray levels. The image was degraded by convolving with a 3×3 blur function as in (10) using circulant boundary conditions; 22 dB white Gaussian noise was added after convolution. A perfect image was obtained after 6 iterations without preprocessing. We set the initial state of all neurons to equal 1, i.e. firing and chose $\lambda = 0$ due to the well conditioning of the blur function.

Figure 2(a) shows the original girl image. The original image is of size 256×256 with 256 gray levels. The variance of the original image is 2797.141. It was degraded by a 5×5 uniform blur function. A small amount of quantization noise was introduced by quantizing the convolution results to 8 bits. The noisy blurred image is shown in Figure 2(b). For comparison purpose, Figure 2(c) shows the output of an inverse filter

[8], completely overridden by the amplified noise and the ringing effects due to the ill conditioned blur matrix H. Since the blur matrix H corresponding to the 5×5 uniform blur function is not singular, the pseudoinverse filter [8] and the inverse filter have the same output. The restored image by using our approach is shown in Figure 2(d). In order to avoid the ringing effect, due to the boundary conditions, we took 4 pixel wide boundaries, i.e. the first and last four rows and columns, from the original image and updated the interior region (248×248) of the image only. The noisy blurred image was used as an initial condition for accelerating the convergence. The constant λ was set to zero because of small noise and good boundary values. The restored image in Figure 2(d) was obtained after 213 iterations. The square error (i.e. energy function) defined in (14) is 0.02543 and the square error between the original and restored imges is 66.5027.

7 Choosing Boundary values

As mentioned in [9], choosing boundary values is a common problem for techniques ringing from deterministic inverse filter algorithms to stochastic Kalman filters. In these algorithms boundary values determine the entire solution when the blur is uniform [10]. In our neural network approach, the same problem occurs. Since the 5×5 uniform blur function is ill conditioned, improper boundary values may cause ringing which may destroy the restored image completely. For example, appending zeros to the image as boundary values introduces a sharp edge at the image border and triggers ringing in

the restored image even if the image has zero mean. Another procedure is to assume a periodic boundary. When the left (top) and right (bottom) borders of the image are different, a sharp edge is formed and ringing results even though the degraded image has been formed by blurring with periodic boundary conditions. The drawbacks of these two assumptions for boundary values were reported in [9,11,12] for the 2–D Kalman filtering technique. We also tested our algorithm using these two assumptions for boundary values, the results indicate the restored images were seriously affected by ringing.

In the last section, to avoid the ringing effect we took 4 pixel wide borders from the original image as boundary values for restoration. Since the original image is not available in practice always, an alternative to eliminate the ringing effect caused by sharp false edges is to use the blurred noisy boundaries from the degraded image. Figure 3(a) shows the restored image using the first and last four rows and columns of the blurred noisy image, Figure 2(b), as boundary values. In the restored image there still exists some ringing due to the naturally occuring sharp edges in the region near the borders in the original image, but not due to boundary values. A typical cut of the restored image to illustrate ringing near the borders is shown in Figure 4. To remove the ringing near the borders caused by naturally occurence sharp edges in the original image, we suggest the following techniques.

First, divide the image into three regions: border, subborder and interior region as shown in figure 5. For 5×5 uniform blur, the border region will be 4 pixels wide due

to the boundary effect of the bias input $I_{i,k}$ in (18), and the subborder region will be 4 or 8 pixels wide. In fact, the width of subborder region will be image dependent. If the regions near the border are smooth, then the width of subborder region will be small or even zero. If the border contains many sharp edges the width will be large. For the real girl image, we chose the width of the subborder region to be 8 pixels. We suggest using one of the following two methods.

Method 1: Adding constraint to subborder region by using blurred image references. In the case of small noise, such as quantization error noise, the blurred image is usually smooth. Therefore, we restricted the difference between the restored and blurred image in the subborder region to a certain range to reduce the ringing effect. Mathematically, this constraint can be written as

$$\|\hat{x}_i - y_i\| \leq T \quad for\ i \in\ subborder\ region, \tag{30}$$

where T is a threshold and $\hat{x}_i$ is the restored image gray value. Figure 3(b) shows the result of using this method with $T = 10$.

Method 2: Adding constraint to the subborder region by using a high pass filter. This method simply sets λ in (14) to zero in the interior region and nonzero in the subborder region, respectively. Figure 3(c) shows the result of using this method with $\lambda = 0.09$. In this case, D was a Laplacian operator.

Experimental results show that both method 1 and 2, using the suboptimal blurred boundary values, reduce the ringing effect significantly.

8 Comparisons with Other Restoration Methods

Comparing the performance of different restoration methods needs some quality measures which are difficult to define owing to the lack of knowledge about the human visual system. The word "optimal" used in the restoration techniques usually refers only to a mathematical concept, and is not related to response of the human visual system. For instance, when the blur function is ill conditioned and the SNR is low, the MMSE method improves the SNR, but the resulting image is not visually good. We believe that human objective evaluation is the best ultimate judgment. Meanwhile, the mean square error or least square error can be used as a reference.

For comparison purposes, we give the outputs of inverse filter, SVD pseudoinverse filter, MMSE filter and modified MMSE filter in terms of the Gaussian Markov random field (GMRF) model parameters [13,14].

8.1 Inverse Filter and SVD Pseudoinverse Filter

An inverse filter can be used to restore an image degraded by a space invariant blur function with high signal to noise ratio. When the blur function has some singular points, an SVD pseudoinverse filter is needed; However both filters are very sensitve to noise. This is because the noise is amplified in the same way as the signal components to be restored. The inverse filter and SVD pseudoinverse filter were applied to an image degraded by the 5×5 uniform blur function and quantization noise (about 40 dB SNR).

The blurred and restored images are shown in Figure 2(b) and 2(c), respectively. As we mentioned before the outputs of these filters are completely overridden by the amplified noise and ringing effects.

8.2 MMSE filter and modified MMSE filter

The MMSE filter is also known as the Wiener filter (in frequency domain). Under the assumption that the original image obeys a GMRF model, the MMSE filter (or Wiener filter) can be represented in terms of the GMRF model parameters and the blur function. In our implementation of the MMSE filter, we used a known blur function, unknown noise variance and the GMRF model parameters estimated from the blurred noisy image by a maximum likelihood (ML) method [13]. The image shown in Figure 6(a) was degraded by 5×5 uniform blur function and 20 dB SNR additive white Gaussian noise. The restored image is shown in Figure 6(b).

The modified MMSE filter in terms of the GMRF model parameters is a linear weighted combination of a Wiener filter with a smoothing operator (such as median filter) and a pseudoinverse filter to smooth the noise and preserve the edge of the restored image simultaneously. Details of this filter can be found in [14]. We applied the modified MMSE filter to the same image used in the MMSE filter above with the same model parameters. The smoothing operator is a 9×9 cross shape median filter. The resulting image is shown in Figure 6(c).

Method	MMSE	MMSE (o)	Modified MMSE	Neural netwoek
Mean square error	1.384 dB	2.139 dB	1.893 dB	1.682 dB

Table 1: Mean square error improvement.

The result of our method is also shown in Figure 6(d). The D we used in (14) was a Laplacian operator as in (15). We chose $\lambda = 0.0625$ and used 4 pixel wide blurred noisy boundaries for restoration. The total number of iterations was 20. The improvement of mean square error between the restored image and the original image for each method is shown in Table 1. In the table the "MMSE (o)" denotes that the parameters were estimated from the original image. The restored image using "MMSE (o)" is very similar to Figure 6(a). As we mentioned before, the comparison of the outputs of the different restoration methods is a difficult problem. The MMSE filter visually gives the worst output which has the smallest mean square error for MMSE(o) case. The result of our method is smoother than that of the MMSE filter. Although the output of the modified MMSE filter is smooth in flat regions, it contains some artifacts and snake effects at the edges due to using a large sized median filter.

9 Conclusion

This paper has introduced a new approach for the restoration of gray level images degraded by a shift invariant blur function and additive noise. The restoration procedure consists of two steps: parameter estimation and image reconstruction. In order to reduce computational complexity, a practical algorithm (Algorithm 2), which has equivalent results to the original one (Algorithm 1), is developed under the assumption that the neurons are sequentially visited. The image is generated iteratively by updating the neurons representing the image gray levels via a simple sum scheme. As no matrices are inverted, the serious problem of ringing due to the ill conditioned blur matrix H and noise overriding caused by inverse filter or pseudoinverse inverse filter are avoided by using suboptimal boundary conditions. For the case of a 2–D uniform blur plus small noise, the neural network based approach gives high quality images whereas the inverse filter and pseudoinverse filter yield poor results. We see from the experimental results that the error defined by (14) is small while the error between the original image and the restored image is relatively large. This is because the neural network decreases energy according to (14) only. Another reason is that when the blur matrix is singular or ill conditioned, the mapping from $\underline{X}$ to $\underline{Y}$ is not one to one, therefore, the error measure (14) is not reliable anymore. In our experiments, when the window size of a uniform blur function is 3×3, the ringing effect was eliminated by using blurred noisy boundary values without any smoothing constraint. When the window size is 5×5, ringing effect

was reduced but not eliminated completely with the help of the smoothing constraint and suboptimal boundary conditions.

References

[1] N. H. Farhat, D. Psaltis, A. Prata, and E. Paek, "Optical Implementation of the Hopfield Model", *Applied Optics*, vol. 24, No.10, pp. 1469–1475, 15 May 1985.

[2] J. J. Hopfield and D. W. Tank, "Neural Computation of Decisions in Optimization Problems", *Biological Cybernetics*, vol. 52, pp. 114–152, 1985.

[3] J. J. Hopfield, "Neural Networks and Physical Systems with Emergent Collective Computational Abilities", *Proc. Natl. Acad. Sci. USA*, vol. 79, pp. 2554–2558, April 1982.

[4] M. Takeda and J. W. Goodman, "Neural Networks for Computation: Number Representations and Programming Complexity", *Applied Optics*, vol. 25, No. 18, pp. 3033–3046, Sept. 1986.

[5] J. P. LaSalle, *The Stability and Control of Discrete Processes*, Springer-Verlag, New York, New York, 1986.

[6] N. Metropolis et al., "Equations of State Calculations by Fast Computing Machines", *J. Chem. Phys.*, vol. 21, pp. 1087–1091, 1953.

[7] S. Kirkpatrick et al., "Optimization by Stimulated Annealing", *Science*, vol. 220, pp. 671–680, 1983.

[8] W. K. Pratt et al, "Visual Discrimination of Stochastic Texture Fields", *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-8, pp. 796–814, Nov. 1978.

[9] J. W. Woods, J. Biemond, and A. M. Tekalp, "Boundary Value Problem in Image Restoration", In *Proc. Intl. Conf. on Acoustics, Speech, and Signal Processing*, pp. 692–695, Tampa, FL, March 1985.

[10] M. M. Sondhi, "The Removal of Spatially Invariant Degradations", *Proc. of IEEE*, vol. 60, pp. 842–853, July 1972.

[11] J. W. Woods and V. K. Ingle, "Kalman Filtering in Two Dimensions: Further Results", *IEEE Trans. Acoust,Speech,Signal Processing*, vol. ASSP-29, pp. 188–197, April. 1981.

[12] J. Biemond, J. Rieske, and J. Gerbrand, "A Fast Kalman Filter for Images Degraded by Both Blur and Noise", *IEEE Trans. Acoust,Speech,Signal Processing*, vol. ASSP-31, pp. 1248–1256, October. 1983.

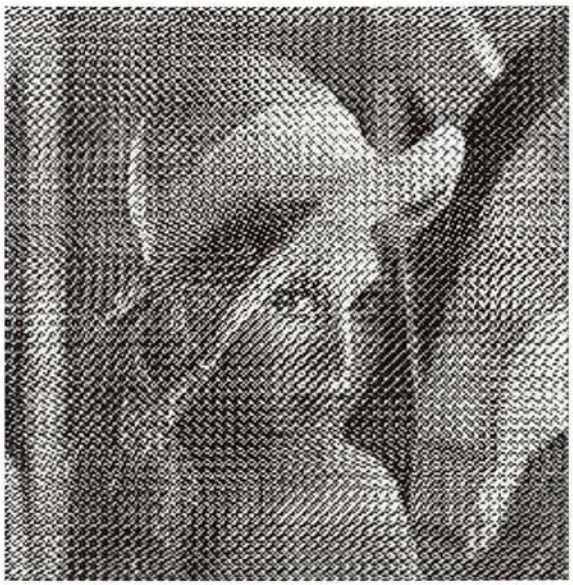
[13] R. Chellappa and H. Jinchi, "A Nonrecursive Filter for Edge Preserving Image Restoration", In *Proc. Intl. Conf. on Acoustics, Speech, and Signal Processing*, pp. 652–655, Tampa, FL, March 1985.

[14] H. Jinchi and R. Chellappa, "Restoration of Blurred and Noisy Image Using Gaussian Markov Random Field Models", In *Proc. Conf. on Information Science and System*, pp. 34–39, Princeton Univ., NJ, 1986.

[15] J. F. Abramatic and L. M. Silverman, "Nonlinear Restoration of Noisy Images", *IEEE Trans. on Patt. Anal. and Mach. Intel.*, vol. PAMI-4, pp. 141–149, March 1982.

(a) Original girl image.

(b) Image degraded by 5×5 uniform blur and quantization noise



(c) The restored image using inverse filter.

(d) The restored image using our approach.

Figure 2: Restoration of noisy blurred real image.

(a) Blurred noisy bounadries.

(b) Method 1.

(c) Method 2.

Figure 3: Results using blurrd noisy boundaries.

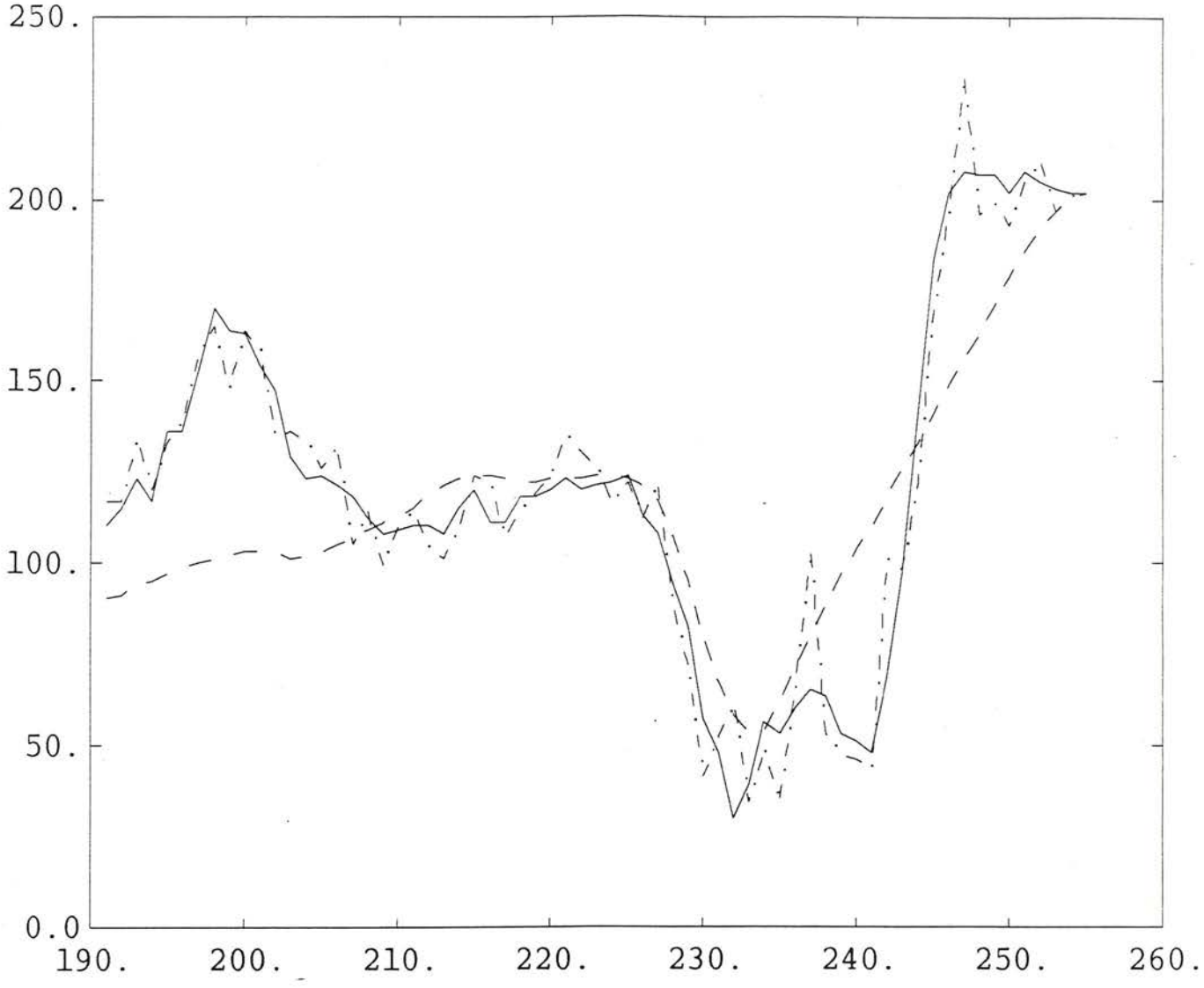


Figure 4: One typical cut of the restored image using the blurred noisy boundaries. solid line for original image, dashed line for blurred noisy image and dashed and dotted line for restored image

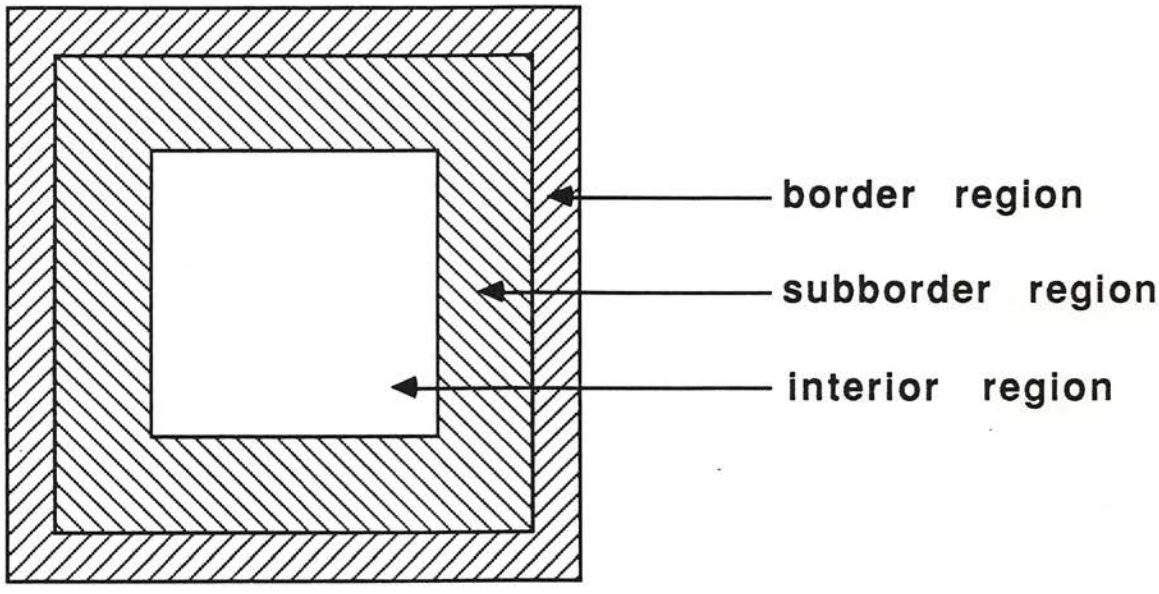


Figure 5: Border, subborder and interior regions of the image.

(a) Image degraded by 5×5 uniform blur and 20 dB SNR additive white Gaussian noise

(b) The rsetored image using the MMSE filter.

(c) The rsetored image using the modified MMSE filter.

(d) The restored image using our approach.

Figure 6: Comparison with other restoration methods.