Image Restoration Using a Multilayer Perceptron with a Multilevel Sigmoidal Function

K. Sivakumar and U. B. Desai

Abstract—**The problem of restoring a blurred and noisy image having many gray levels, without any knowledge of the blurring function and the statistics of the additive noise, is considered. A multilevel sigmoidal function is used as the node nonlinearity, due to which the same number of nodes as in the case of a binary image is sufficient for an image with multiple gray levels. Restoration is achieved by exploiting the generalization capabilities of the multilayer perceptron network. For realistic images, training time becomes a major burden. To overcome this, a segmentation scheme is suggested. Simulation results are also provided.**

I. Introduction

Consider the image observation model

$$y(i, j) = \sum_{k,l \in S} h(k, l)x(i - k, j - l) + n(i, j); \quad i, j = 0 \cdots (N - 1) \tag{1}$$

where S denotes the convolution window, $h(.,.)$ is the blurring function, $x(.,.)$ is the original image, $y(.,.)$ is the recorded image, and $n(.,.)$ is the additive noise. Using a lexicographical ordering, the above can be expressed in matrix form as

$$\boldsymbol{Y} = \boldsymbol{HX} + \boldsymbol{N} \tag{2}$$

where $\boldsymbol{H}$ is an $N^2 \times N^2$ blur matrix corresponding to the blur function $h(.,.)$, $\boldsymbol{X}$ and $\boldsymbol{Y}$ are $N^2 \times 1$ vectors corresponding to $x(.,.)$ and $y(.,.)$, respectively, and $\boldsymbol{N}$ is a $N^2 \times 1$ vector corresponding to $n(.,.)$.

Conventional image restoration methods, like the ones using an inverse filter, Wiener filter, or Kalman filter, assume that $h(.,.)$ is exactly known and the statistics of $n(.,.)$ are also known [1]–[3]. In a realistic situation, neither of these assumptions would hold true and consequently some form of adaptive scheme would have to be used. Recently, there has been some work along these lines, see for example [4], [6]–[8]; but here too the noise statistic is assumed known, or the noise distribution is assumed with possibly unknown parameters.

With the advent of artificial neural networks (ANN), and the potentials offered by it, there has been some work on restoration of images using ANN. Most notable is the work of Zhou *et al.* [9] where a Hopfield network is used for restoration. Recently we became aware of the paper by Paik and Katsaggelos [10] which proposes a modified Hopfield neural network for regularized image restoration. Also [11] contains chapters (for example [12]) dealing with the use of a Hopfield type neural network for image restoration. These methods again assume that the blurring matrix $\boldsymbol{H}$ is known. Moreover, the number of nodes in the network is directly proportional to the number of gray levels in the image. References [9], [10] have provided the motivation for our work.

Manuscript received August 21, 1991; revised July 17, 1992. The associate editor coordinating the review of this correspondence and approving it for publication was Prof. S. Y. Kung.

K. Sivakumar is with the Department of Electrical and Computer Engineering, Johns Hopkins University, Baltimore, MD 21218.

U. B. Desai is with the Department of Electrical Engineering, Indian Institute of Technology, Powai, Bombay 400076, India.

IEEE Log Number 9207547.

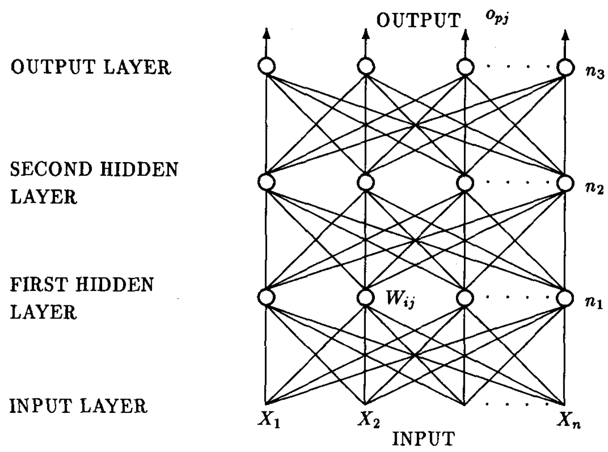

Fig. 1. A three-layer perceptron.

In this correspondence, we consider the problem of restoring $\boldsymbol{X}$ from $\boldsymbol{Y}$, without any knowledge of the blurring function $\boldsymbol{H}$ and the statistics of the additive noise $\boldsymbol{N}$. If only a binary image is considered, then by exploiting generalization capabilities, the standard multilayer perceptron (MLP) network (Fig. 1) can be trained to perform image restoration. Nevertheless, any real image would have multiple gray levels. In such a situation, a common approach to extend the results of binary images is to consider one plane of neurons corresponding to each bit in the gray level representation. Moreover, these planes are interconnected. Thus there is an explosion in the number of neurons used. Our approach is to use an MLP network having the same order of neurons as required for the binary image, but with a modified sigmoidal function to tackle the problem of multiple gray levels. Such a modified sigmoidal function is shown in Fig. 2 for the case of four gray levels and is referred to as a multilevel sigmoidal function (MSF).

From (3) it is clear that the MSF is also differentiable, and consequently the back propagation (BP) algorithm is easily extendable. Details of this are presented in Section II. In Section III, the MLP based image restoration strategy is presented along with simulations. As mentioned earlier, the generalization capability of the MLP network is exploited to achieve restoration. During simulations we observed that for "reasonable" generalization, the MLP had to be trained for about 10% of the possible images in the given class. This number becomes prohibitively large for realistic images and thereby exponentially increases the training time. To overcome this problem we adopt a segmentation strategy. This is described in Section IV.

II. Multilevel Sigmoidal Function

Consider a three-layer perceptron shown in Fig. 1, with the q-level sigmoidal nonlinearity,

$$f(x) = \sum_{i=-(q/2-1)}^{(q/2-1)} \left(\frac{1}{1 + \exp(-x + 8i)} \right). \tag{3}$$

A four-level sigmoidal function is shown in Fig. 2. In the definition of $f(x)$, the number 8 is used so that the standard sigmoidal functions are shifted sufficiently, and add up to give a multilevel sigmoidal function with distinct levels. The shift is also not excessive,

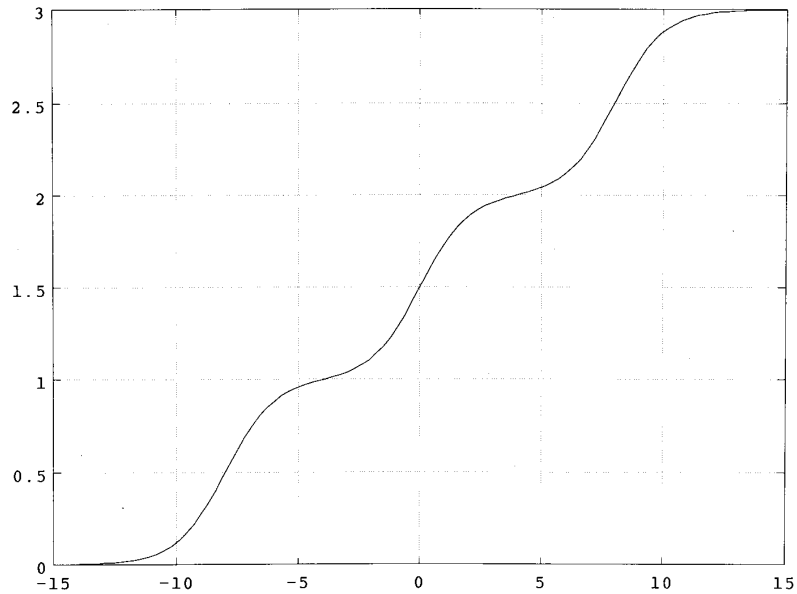

Fig. 2. Modified sigmoidal function.

which would make the derivative of $f(x)$ very small near the different gray levels, thereby decreasing the rate of convergence. Differentiability of $f(x)$ is obvious, indeed

$$\frac{df}{dx} \triangleq f'(x) = \sum_{i=-(q/2-1)}^{(q/2-1)} \left(\frac{1}{1+\exp(-x+8i)}\right) \cdot \left(1 - \frac{1}{1+\exp(-x+8i)}\right). \tag{4}$$

The back propagation algorithm [13] is obtained by minimizing the mean-square error

$$E_p = \frac{1}{2} \sum_j (t_{pj} - o_{pj})^2 \tag{5}$$

where t_{pj} and o_{pj} are the desired and actual output of node j when pattern p is presented. The sum of the inputs at a node j in any layer when the pattern p is presented to the network is

$$\text{net}_{pj} = \sum_i W_{ij} o_{pi} \tag{6}$$

where

$$o_{pi} = f(\text{net}_{pi}) \tag{7}$$

is the output of node i of the previous layer. The interconnection weights are updated using the principle of steepest descent,

$$W_{ij}(t+1) = W_{ij}(t) - \eta \frac{\partial E_p}{\partial W_{ij}(t)} \tag{8}$$

where η is referred to as the learning rate. Carrying out the necessary algebra one obtains the following updating equations:

$$W_{ij}(t+1) = W_{ij}(t) + \eta \delta_{pj} x_i \tag{9}$$

where

i) For a node j in the output layer,

$$\delta_{pj} = (t_{pj} - o_{pj}) f'(\text{net}_{pj}) \tag{10}$$

ii) For a node j in the hidden layer,

$$\delta_{pj} = \sum_k \delta_{pk} W_{jk}(t) f'(\text{net}_{pj}). \tag{11}$$

Convergence is sometimes faster when a momentum term α is added to the above equation:

$$W_{ij}(t+1) = W_{ij}(t) + \eta \delta_{pj} x_i + \alpha(W_{ij}(t) - W_{ij}(t-1)). \tag{12}$$

Remarks:

1) One view about the use of multilevel sigmoidal function is that the number of neuron complexity is replaced by a more complex node transfer function.

2) The proposed multilevel sigmoidal function is more stable near the different gray levels than a simple scaled sigmoidal function given by $(q-1)(1/(1+\exp(-x)))$, where q represents the number of gray levels.

III. The MLP Model for Image Restoration

The basic scheme is represented in the form of a block diagram in Fig. 3. We consider an image $x(.,.)$ with q gray levels, degraded according to (1). A three-layer perceptron is then trained with a small subset of all the possible images $x(.,.)$. For $N \times N$ images there are N^2 input units and N^2 output units. During the training process the actual output of the network is computed by feedforward and compared with the desired output. The resulting output is used in the BP algorithm (3)–(12) to change the interconnection weights. It is assumed that the degradation process undergone by all the images is the same, i.e., $\boldsymbol{H}$ is the same and the statistics of $\boldsymbol{N}$ remain the same. Such an assumption would hold true, for example, in the case of images transmitted through a particular channel or recorded by the same image forming system. It is expected that if the training set is sufficiently large the network would restore those images (which it has not been trained with) that have undergone similar type of degradation. The conditions for valid generalization by a network are reported by Baum and Haussler in [14]. A copy of this paper was not available and hence

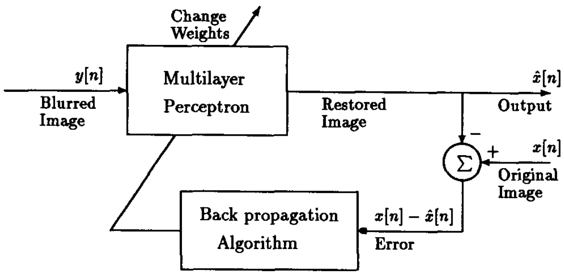

Fig. 3. Multilayer perceptron model for image restoration.

our work, to a large extent, is empirical. We found that for reasonable restoration, the network had to be trained with approximately 10% of all possible images. The network is then presented with the degraded image and the output of the network gives the restored image. Thus, an explicit model of the blurring function or the noise statistics is not required.

Simulation

To reduce the computational complexity we initially used 3×3 binary images during our simulations. A set of 50 images (chosen randomly from the possible 512 images) were used for training. The remaining 462 images were used for testing the network for generalization, after training. The network configuration used was 9-15-9-9, i.e., 9 input nodes, 15 nodes in the first hidden layer, 9 nodes in the second hidden layer, and 9 output nodes. It was observed that for generalization, the total number of weights and thresholds (variables) in the network had to be equal to, or slightly less than, the number of images to be trained multiplied by the size of the images. For example, in our case, the network consisted of 351 weights and 33 thresholds, a total of 384 variables. The training set consisted of 50 images of 9 inputs each, a total of 450. The learning rate and the momentum term in the BP algorithm were chosen to be $\alpha = 0.2$ and $\eta = 0.2$. During training, an average error of 0.035 in each output was tolerated. For the test images an output of 0.8 or more was taken to be 1 and an output of 0.2 or less was taken to be 0.

During the first set of simulations, the images were degraded with a blurring function alone. The blurring process can be represented as

$$Y = HX \tag{13}$$

where, $\boldsymbol{H}$ is block circulant blur matrix corresponding to a blur function; $\boldsymbol{X}$ and $\boldsymbol{Y}$ are the original and degraded images, respectively. The shift-invariant blur function that was used can be written as a convolution over a small window as

$$\frac{1}{16}\begin{bmatrix} 1 & 1 & 1 \\ 1 & 8 & 1 \\ 1 & 1 & 1 \end{bmatrix}.$$

Fig. 4(a) shows the number of images that were restored (other than those trained) without any error, with error in one pixel and with error in two pixels, as the number of images in the training set is increased. During the second set of simulations, the images were first blurred as before, then zero-mean white Gaussian noise was added to the images. The simulations were carried out for two different values of signal-to-noise ratio (SNR), 15 and 10 dB, where $\text{SNR} = 10 \log \sigma_s^2/\sigma_n^2$; σ_s^2 and σ_n^2 are, respectively, signal and noise variance. Figs. 4(b) and (c) show the simulation results for the two cases. A comparison of the performance of the network for the three different cases considered above is given in Fig. 4(d).

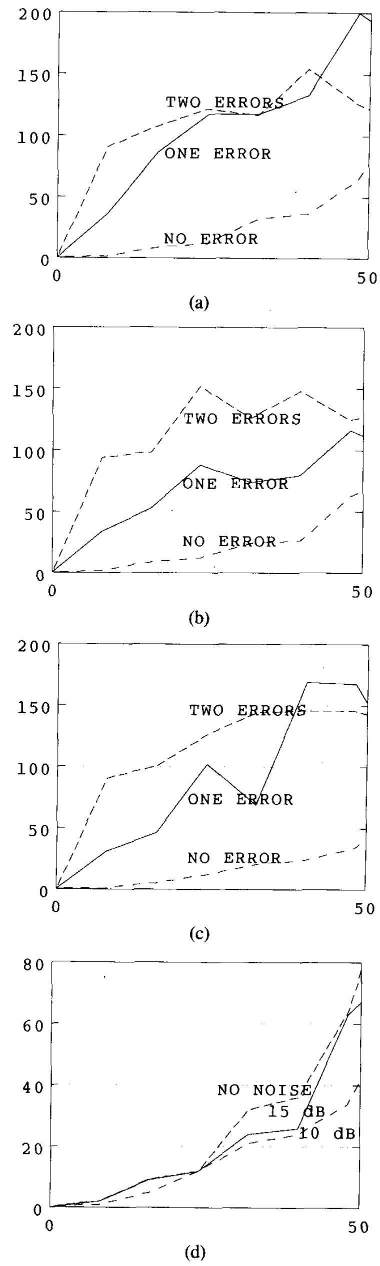

Fig. 4. Variation of the number of images restored (other than those trained) with respect to the number of images trained. Images were blurred according to (2): (a) without noise; (b) with a SNR of 15 dB; (c) with a SNR of 10 dB; and (d) a comparison of the three different cases.

From the simulations, it can be observed that as the number of images used for training is increased, the generalization is better. The number of images restored increases sharply after the network has been trained for about 40 images. The generalization is substantial when the network is trained for approximately 10% of the total number (512) of images. In a few cases, changing the tolerance during training to 0.01 increased the number of images restored. It was further observed that the trained network for the case of a SNR of 15 dB could restore more images than the network trained for zero noise, when presented with images with and without noise.

Later, we considered 3×3 images with four gray levels. Since even 1% of the total possible set of images is more than 2500, a complete set of simulations for the case of four gray levels was not possible with the available computing facilities. A set of 200 3×3 images with four gray levels were randomly generated and 30 of these were used for training the network. After training, the network could successfully restore one of the remaining 170 images without any error and three images with an error of one level in a single pixel. The remaining images were restored with an error of one level in two or three pixels. For large images with a number of gray levels, this error in a few pixels may not be significant visually. It is expected that as the size of the training set is increased, the generalization done by the network would be significant.

IV. Segmentation Scheme for Large Patterns

Since for proper generalization, the network has to be trained for at least 10% of the total number of images, the number of images to be trained would increase exponentially with the size of the image. Thus, if we have an $N \times N$ image with q gray levels, the training set should have at least 10% of q^{N^2} images. This would increase the training time tremendously. Hence, a direct implementation of this method for large images is not feasible. We now suggest a segmentation scheme with which it is possible to use this method for large images.

The method involves segmenting a large image into smaller subpatterns as shown in Fig. 5. It is assumed that the blurring undergone by the image is only local, i.e., each pixel is affected by only its neighboring pixels. Hence an estimate of the image can be made from the information about a few neighboring pixels alone. Let us consider an $N \times N$ image with circulant boundaries. It is segmented into $m \times m$ subpatterns such that two adjacent subpatterns in the same row (column) overlap in $(m - 1)$ columns (rows). There will be $N \times N$ such subpatterns such that each pixel is at the center of a particular sub-pattern. Each subpattern is fed as input to a network that has been trained for $m \times m$ images. By taking the output of the node corresponding to the center pixel of the subpattern from each network the image can be reconstructed. Training different networks with different sets of images could improve the overall performance. One could have $N \times N$ such subnetworks operating in parallel or only one network which could be sequentially fed with the different subpatterns. Since the restoration time for a single subpattern is very small, this does not drastically increase the total time taken. In more complex segmentation schemes, the overlap between adjacent subpatterns can be decreased. The output of all the nodes, weighted appropriately, can then be used for reconstructing the image.

The first segmentation scheme described above was implemented for a number of 16×16 and a few 32×32 binary images with $m = 3$. These images were degraded by a blur function alone. It was observed that on an average, the restored image was in error in 22% of the pixels. Further, when the network used for restoring each subpattern was trained with images degraded by blur function and noise (SNR of 15 dB), the error in the final image was in only 16% of the pixels. Thus, again the network trained for a SNR of 15 dB seems to perform better than the network trained for zero noise. By reducing the amount of blur, there was no significant

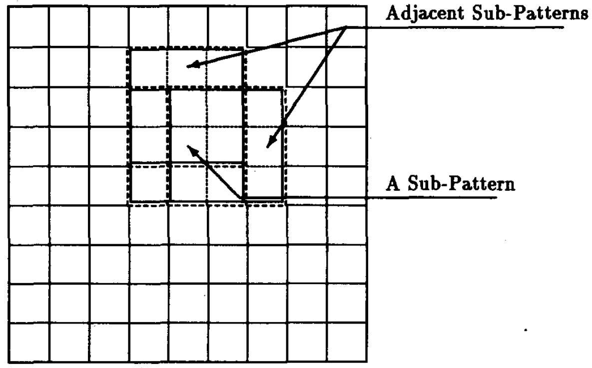

Fig. 5. Segmentation scheme.

decrease in the number of pixels in error. Hence the errors in the restored image seem to be due to the improper generalization of the network and not because of segmentation. This can be minimized by training the network with more number of images and decreasing the tolerance during training.

V. Conclusions

A restoration scheme for images with multiple gray levels using a multilevel sigmoidal function has been developed. This modified node transfer function incorporates the complexity due to the increase in the number of gray levels. The method makes use of the generalization capability of the MLP. The method is robust in the sense that it does not require any information about the degradation process. To reduce the complexity of the problem for large images, a segmentation scheme has been proposed. At present we are investigating the applicability of this scheme for nonlinear blurring and multiplicative noise corruption.

References

[1] W. K. Pratt, *Digital Image Processing*. New York: Wiley, 1978.

[2] R. C. Gonzalez and P. Wintz, *Digital Image Processing*. Menlo Park, CA: Addison-Wesley, 1987.

[3] A. K. Jain, *Fundamentals of Digital Image Processing*. Englewood Cliffs, NJ: Prentice-Hall, 1989.

[4] Y. S. Boutalis, S. D. Kollias, and G. Carayannis, "A fast multichannel approach to adaptive image estimation," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 37, pp. 1090–1098, 1989.

[5] J. S. Lim, *Two-Dimensional Signal and Image Processing*. Englewood Cliffs, NJ: Prentice-Hall, 1990.

[6] M. M. Hadhoud and D. W. Thomas, "The two-dimensional adaptive LMS (TDLMS) algorithm," *IEEE Trans. Circuits Syst.*, vol. 35, pp. 485–494, 1986.

[7] A. Tekalp, H. Kaufman, and J. Woods, "Fast recursive estimation of the parameters of space varying AR images," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-33, pp. 469–471, 1985.

[8] U. B. Desai, Z. D. Dittia, P. S. Kumar, and P. Y. Mundkur, "Sequential and parallel schemes for adaptive 2-D parameter estimation with application to image estimation," *Sadhana*, vol. 15, part 3, pp. 213–234, Nov. 1990.

[9] Y. T. Zhou, R. Chellappa, A. Vaid, and B. K. Jenkins, "Image restoration using a neural network," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-36, pp. 1141–1151, 1988.

[10] J. K. Paik and A. K. Katsaggelos, "Image restoration using a modified Hopfield network," *IEEE Trans. Signal Processing*, to be published.

[11] A. K. Katsaggelos, Ed., *Digital Image Restoration*, Springer Series in Information Sciences, vol. 23. Springer-Verlag, 1991.

[12] S. J. Yeh, H. Stark, and M. I. Sezan, "Hopfield-type neural networks: Their set-theoretic formulation as associative memories, classifiers, and their applications to image restoration," in *Digital Image Restoration*, A. K. Katsaggelos, Ed., Springer Series in Information Sciences, vol. 23. Springer-Verlag, 1991, ch. 3.

[13] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning internal representations by error propagations," in *Parallel Distributed Processing Explorations in the Microstructure of Cognition*, vol. 1, *Foundations*, D. E. Rumelhart and J. L. McClelland, Eds. Cambridge, MA: M.I.T. Press, 1986.

[14] E. Baum and D. Haussler, "What size nets give valid generalization," *Neural Computation*, vol. 1, pp. 151-160,1989.

CORRECTION

A Local Update Strategy for Iterative Reconstruction from Projections

K. Sauer and C. Bouman

This paper may be found on pages 534 to 548 in the February 1993 issue. The title was wrongly listed on the cover under "Underwater Acoustics Signal Processing." It should have been listed under "Multidimensional Signal Processing."