Neural-network-aided design for image processing

Ilia Vitsnudel, Ran Ginosar, Yehoshua Zeevi

Ilia Vitsnudel, Ran Ginosar, Yehoshua Y. Zeevi, "Neural-network-aided design for image processing," Proc. SPIE 1606, Visual Communications and Image Processing '91: Image Processing, (1 November 1991); doi: 10.1117/12.50370

SPIE.

Event: Visual Communications, '91, 1991, Boston, MA, United States

Neural Network Aided Design for Image Processing

Ilia Vitsnudel, Ran Ginosar and Yehoshua Zeevi

Technion - Israel Institute of Technology
Department of Electrical Engineering
Haifa, Israel, 32000

ABSTRACT

A new concept of Neural Network - Aided Design (NN-AD) is presented. It is a hierarchical approach consisting of several concatenated stages of visual information processing which are designed by training Neural Networks (NN). Thus, NN-AD can be viewed as a general tool for the design of special filters in accordance with the specific task of image processing under consideration. The nonlinear filters are formatted by a supervised presentation of a proper set of input-output patterns. The principles of NN-AD design are illustrated by the examples of edge detection with subpixel resolution and of orientational processing for edge enhancement. The proposed NN-AD approach is found to be very robust with regard to various types of errors.

1. INTRODUCTION

A variety of algorithms for image processing sometimes require very tedious implementation in computer software. Moreover, a direct application of specially-designed procedures for feature enhancement often generates artifacts. This happens because of processing on improper neighborhood for the purpose of emphasizing a certain feature. In this paper we propose a conceptual framework, designated *Neural Network-Aided Design* (NN-AD), for performing various tasks of image processing.

Neural Networks (NN) have been widely applied in image processing[1-5]. The majority of applications utilize a Hopfield-type NN structure[1,2], while others use derivatives of multi-layer feed-forward NN architectures to execute various classification and recognition tasks[3,4]. Some studies exploit the ADALINE[4] and multi-layer perceptrons[5], to design local image processing tools. In the present study we propose a much broader application of NN (especially of multi-layer feed-forward NN) for the design of a class of non-linear filters based on the appropriately acquired training sets.

The organization of the paper is as follows. In the next section we present an example of a direct application of multi-layer feed-forward NN to the problem of edge detection with subpixel accuracy. This example illustrates the methods of creating a set of training patterns for NN. Then, by presenting a procedure for designing an edge restoration processor, it is demonstrated that a direct manual approach to filter design results in artifacts. Subsequently, we propose how to resolve this problem with the NN-AD approach and discuss the advantages offered by the new method.

2. FIRST EXAMPLE

To introduce the concept of NN-AD we present an example in which a processor for edge detection with subpixel precision is constructed. A multi-layer feed-forward NN architecture (with one hidden layer) is employed to construct a non-linear filter which operates by sliding over the image in a way that is similar to convolution with a linear FIR. The network is presented with a set of input-output patterns, obtained in the following way: an original high-resolution training image (512x512) is used first for edge detection by means of a modified scheme for estimation of the second directional derivative, described by Haralick[6]. In addition to this scheme, a weighted estimation of polynomial coefficients is used in order to increase the contribution of closer-to-center pixels in the working neighborhood. The results of this procedure represent the output patterns. Next, images to be used as input patterns are obtained by decimation of the original images of higher

0-8194-0743-7/91/$4.00

resolution in one of two ways, each simulating one of two alternative methods of the low-resolution image acquisition, either by deleting the neighboring pixels, or by reducing the resolution by means of a Gaussian filter (i.e., equivalent to weighted averaging of neighboring pixels).

The training strategy is based on minimization of the appropriate mean square error criterion:

$$\min\|y - H(x)\| \tag{1}$$

where y is the desired edges output pattern of the high-resolution image, H is the general description of NN operation and x is the low-resolution input pattern of the same image. Upon the convergence of the training procedure, achieved by the standard Back-Propagation algorithm, we obtain the optimal network capable of predicting edge positions on the low-resolution image with subpixel accuracy. This procedure may be fine-tuned to the particular situation, i.e. particular kind of images, format and method of the image acquisition and the desired factor of subpixel accuracy (resolution). Also, the method for edge evaluation in high-resolution training may be selected in accordance with the situation. One of the alternatives for obtaining the set of input-output training patterns is to create a pool of artificial images. These images, consisting of mostly geometrical structures with precisely known position of edges, may be used for training the network in conjunction with the naturally created images as described before.

Figure 1 illustrates the result of application of such trained NN to test images in the detection and interpolation of edges with subpixel resolution. Figure 1a depicts the original low-resolution image (256x256), whereas figure 1b shows the detected edges at high-resolution (512x512).

The above approach offers a simple alternative to the design of a non-linear filter for image processing. This is accomplished without being concerned with the limitations imposed by the implementation of a particular mathematical formalism, as is the case with most other methods. Rather than developing the algorithm and subsequently realizing it by means of convolution masks for the real-time implementation, the general framework of NN-AD described here is used by constructing a suitable set of the training patterns and subsequently applying the training procedure. Having illustrated the proposed approach by an example of application of NN-AD design, we assert that many goals of image processing should be accomplished in an iterative manner based on the tools derived immediately from the images themselves, rather than by manual filter design intended to improve specific image features.

In order to clarify this, we present in the following section an example of a filter which is designed for edge restoration in a conventional manner and generates such artifacts. The effects of compression and subsequent decompression by Block List Transform[7] (BLT), or of image magnification by pixel replication and subsequent low-pass filtering, are demonstrated. We then propose the alternative approach using NN-AD.

3. THE EFFECTS OF CONVENTIONAL DESIGN

The underlying principle of the procedure for design of filters for edge restoration is based on the observation that edges are characterized by a higher spectral content in the orthogonal orientation. Thus, a suitable filter for edge improvement should be elliptic in the frequency domain, oriented with its major axis along the direction of maximum gradient. In order to account for different orientations of the edges, we discretize the angular orientation into 16 directions (over the range of 0-180^0) and obtain 16 masks applying the inverse DFT. These masks are applied to the degraded (by BLT) image of figure 2a in accordance with the direction of gradient. The result is shown in figure 2b. It may be easily observed that the edges in the resulting image are drastically improved, but a strong artificial effect is introduced into the image. A large number of speckled irrelevant points is created along the edges of the image. These artifacts are a byproduct of the inaccurate manual design of the edge-restoring filters. They are generated in the regions where filter masks are activated improperly as dictated by the respective value of the gradient. The scaling of different filters also contributes to this behavior.

Figure 1a Low-resolution input image of Lena.

Figure 1b Extrapolated edges location with subpixel precision.

Figure 2a Degraded by BLT compression/decompression image of Lena.

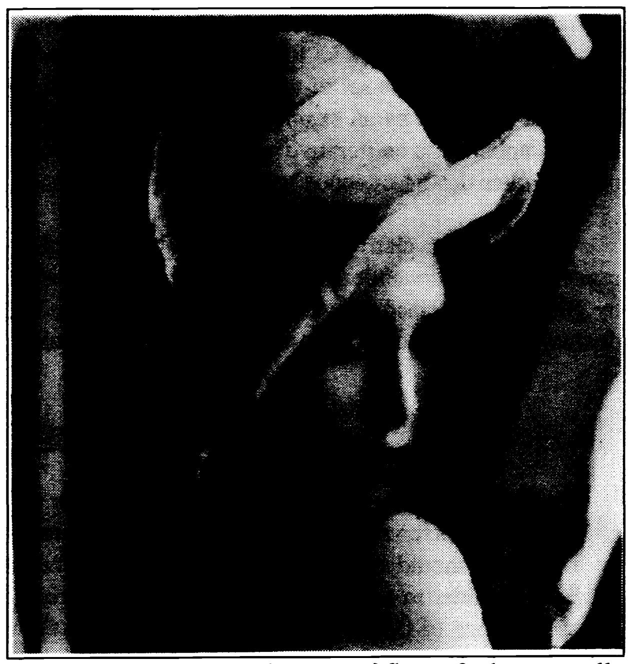

Figure 2b Edge-restored image of figure 2a by manually created directional filter masks. Strong artifacts are present in the form of speckle noise along the edges.

4. NEURAL NETWORKS FOR FEATURE ENHANCEMENT

As demonstrated in the example of the previous section, the manual filter design introduces image artifacts. We therefore devise image processing tools, which are derived from the images by means of a proper training procedure. To highlight this approach we focus again on the example of edge restoration after the degradation by BLT.

To improve edges in the image, we design a special NN which is constrained to generate an output free of undesired properties. (In our example we are interested in images with high quality edges, free of jaginess effects).

The approach is to use a standard learning strategy based on minimization of $\|y-H(x)\|$ as in (1), with an additional term describing the undesired feature: $\min\{\|y-H(x)\|+\alpha\Omega(H(x))\}$, where $\Omega(H(x))$ is a term which emphasizes the undesired features of the image. One particular choice is to set Ω as $\nabla^2 H$, thus imposing a smoothness constraint on the output image, by minimizing second derivatives and thereby suppressing abrupt signal changes. The imposition of this constraint has been dealt with analytically, and the necessary mathematical formalism for the network training algorithm has been developed[8]. Here we prefer to deal with the implications of Ω in an alternative way, namely, by exposing the network to the restricted training set of output signals which exclude the undesired qualities dictated by Ω.

The training strategy is based upon the minimization of (1). The network is presented with selected training patterns consisting, at the input, of edges of the degraded image (with strong BLT jaginess effect), and at the output of the edges of the original undistorted image. Edges are defined in our simulation as the loci where the majority of gradients exceeding some given threshold in the working patch are oriented in the same direction. Thus this procedure applies to fine lines as well.

The network paradigm is shown in figure 3. The network consists of 16 directional and one general subnetworks. Each one of the 16 subnetworks is responsible for the properly oriented neighborhood of the input image, determined by the gradients prevalence. The general unit treats the neighborhoods which have not been classified as oriented. The switching of subnetworks is done through the additional NN, which has been trained separately to exhibit the predominant orientation of a group of pixels within a given neighborhood. At the training stage each directional subnetwork is presented with the properly oriented edges, detected in the undistorted image. The end of the training procedure is established by the convergence of mean output errors of all subnetworks over all images included in a training set. These images comprise natural scenes as well as artificially created ones with dominant angular features. During the training stage, a sliding neighborhood is used as an input pattern.

Optimal parameters of the network have been found empirically as follows: the size of the neighborhood is a patch of 7x7 pixels, and the number of the hidden units in each subnetwork is 8. This provides a reasonable trade-off between the size and speed of calculations.

The resulting procedure enhances features without introducing significant artifacts. To further improve the performance, the network is operated in an iterative manner. The operation of the network may be considered as a kind of "projecting" the degraded image into the space of images with the number of desired properties. Thus, a number of networks (each one designed to enhance a specific image feature) are connected in a serial manner, and the entire process is repeated several times, until the desired effect is sufficiently apparent in the output image. Figure 4 shows results of this procedure applied to restore degraded edges of the image shown in figure 2a. Note that the resulting image does not suffer from the speckle noise along the edges.

5. DISCUSSION

We demonstrated by examples an alternative approach to the traditional manual design of filters for image processing. This approach is based on the application of the NN-AD approach, which may be viewed as a universal tool trainable for the execution of a specific design task. The desired features can be enhanced by formatting a proper NN which is regarded as a special non-linear filter, by exposing the neural network at the training stage to the input-output pairs of images depicting the desired properties.

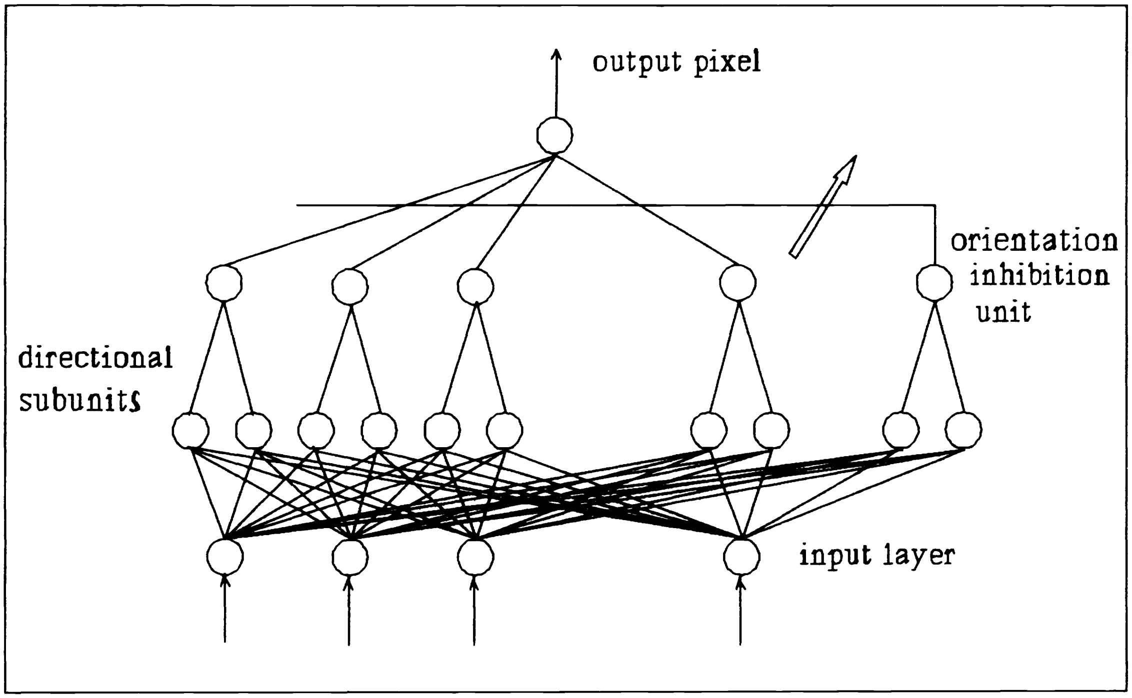

Figure 3 Directional NN paradigm for edge restoration.

Figure 4 Lena image restored by directional NN.

Two examples have been given of this approach. In the first example, the network for edge detection with subpixel accuracy has been constructed based on the multi-layer feed-forward NN paradigm. The high-resolution edge location served as an output of a training pattern corresponding to an input of low-resolution image. Flexibility of the NN-AD enables easy adaptation to different image acquisition methods, image resolutions and formats. The second example has presented a network for edge restoration in degraded images (for example by BLT). The network consists of 16 directional subnetworks and one general subnetwork for representing low gradient values of "unoriented" pixels. This network has been developed in order to overcome problems associated with the manual filter design, which produces serious artifacts in the form of speckle noise along edges, due to erroneous filter operation on the improper neighborhoods. In contrast, the NN derived directly from undistorted training images does not suffer from this effect, being naturally tuned to the images, but may produce much weaker restoration effect. In such cases the network is operated iteratively to arrive at the desired results.

Additional successful implementations of image processing tools bring to the conclusion that NN-AD based on general form NN with appropriately designed set of training patterns provide an attractive approach to the design of special filters for feature enhancement in image processing.

6. REFERENCES

1. J. K. Paik, "Image Restoration and Edge Detection Using Neural Networks," Ph.D. thesis, Nortwestern University, Department of Electrical Engineering and Computer Science, June, 1990.

2. Y. Zhou, R. Chellappa, A. Vaid, and B. Jenkins, "Image Restoration Using a Neural Network," *IEEE Trans. on ASSP*, vol. ASSP-36, pp. 1141-1151, July, 1988.

3. I. C. Jou, "A Neural Network-Based On-Line Chinese Character Recognition System," *J. of Visual Communication and Image Representation*, vol. 2, No. 1, pp. 85-90, March, 1991.

4. H. McCauley, "Target Cuing - a Heterogeneous Neural Network Approach," *Proc. of SPIE Visual Communication and Image Representation 90*, vol. 1360, pp. 27-35, 1991.

5. N. Dodd, "Multi-layer Perceptron Inversion Applied to Image Neighborhood Operations," *Proc. of the IEEE First Int. Conf. on Neural Networks: Vol.IV*, pp. 293-300, San Diego: IEEE, 1987.

6. R. M. Haralick, "Digital Step Edges from Zero Crossing of Second Directional Derivatives," *IEEE Trans. on PAMI*, vol. PAMI-6, pp. 58-68, January, 1984.

7.B. G. Haskell, "Image Coding by the Block List Transform," *AT&T Bell Laboratories Tech. Memo no.11350-850903-01TM*, Sept. 1985.

8. T. Poggio and F. Girosi, "Networks for Approximation and Learning," *Proc. IEEE*, vol. 78, pp. 1481-1497, September, 1990.