Self-Organizing Neural Network Character Recognition on a Massively Parallel Computer

C L. Wilson, R. A. Wilkinson, and M. D. Garris, Advanced Systems Division
National Institute of Standards and Technology, Gaithersburg, MD 20899

International Joint Conference on Neural Networks
Proceeding, **II**, pp. 325-329, June, 18 1990, San Diego, CA

Abstract

Two neural network based methods are combined to develop font independent character recognition on a distributed array processor. Feature localization and noise reduction are achieved using least squares optimized Gabor filtering. The filtered images are then presented to an ART-1 based learning algorithm which produces self-organizing sets of neural network weights used for character recognition. Implementation of these algorithms on highly parallel computer with 1024 processors allows high speed character recognition to be achieved in 3ms/image with greater than 99% accuracy on machine print and 80% accuracy on unconstrained hand printed characters.

1. Introduction

Neural network methods show great promise for providing high accuracy, noise resistant, parallel algorithms and architecture for image recognition. One specific area of image recognition, the conversion of images of hand written and machine print characters to computer representation, has been studied in detail. Both special purpose hardware [1] and entirely software [2] approaches have been used on the character recognition problem with promising results. The present work addresses the problem of using a specific class of computer architecture, an array of 1024 processors arranged in a 32 by 32 grid and operated in a parallel mode [3], as a neural network character recognition device.

In the character recognition operation, images of individual characters from either hand or machine print are recognized as specific characters or are rejected. An isolated character image is presented to the recognition device and a specific character identity or set of identities with associated confidence factors is returned. For a character recognition technology to be of commercial interest, both the accuracy and the ability to detect low confidence recognitions must be present. If human intervention is required to detect failures or if the failure rate is too high existing data entry methods will be more cost effective.

In section 2 the construction of adaptive least squares optimal filters which use matrices of Gabor functions as independent basis functions is discussed. John Daugman has used Gabor functions for image compression and image texture analysis[4]. The motivation for use of incomplete functions, which add considerable complexity to the representation problem, is discussed in detail in [4]. Carpenter and Grossberg [5,6] have described three levels of self-organizing Adaptive Resonance Theory (ART) pattern classification architecture. In ART-1 resonant bottom-up and top-down filters are used to learn and classify binary data [5]. In section 3 the construction a parallel matrix based implementation of the ART-1 method is discussed. In section 4 the assignment of ART-1 weights to specific character classes and the use of a separate learning and classification pass with an ART-1 method are discussed. In section 5 results of character classification experiments with varying amounts of noise and with different combinations of filters are discussed.

2. Least Squares Filtering

The filtering section of the process of recognition is accomplished using a least squares fit, on an image by image basis, of each image. The kernel functions used are Gabor functions. Adopting the convention that bold upper case variables represent array processor matrix data types and bold lower case variables represent array processor vector data types, these are defined as:

$$\mathbf{G}_j(\mathbf{X}, \mathbf{Y}) = \exp(-\mathbf{R}^2) \left\{ \begin{array}{l} \sin(\omega_j \mathbf{X}') \\ \cos(\omega_j \mathbf{X}') \end{array} \right.$$

where the matrix variables $\mathbf{R}$, $\mathbf{X}'$, and $\mathbf{Y}'$ are given by:

$$\mathbf{R}^2 = (\mathbf{X}'^2 + \mathbf{Y}'^2)/\sigma_j^2$$

$$\begin{pmatrix} \mathbf{X}' \\ \mathbf{Y}' \end{pmatrix} = T \begin{pmatrix} \mathbf{X} - x_{0j} \\ \mathbf{Y} - y_{0j} \end{pmatrix}.$$

A typical scalar transformation to be applied to each element of the matrix variables is a rotation of the form:

$$T = \begin{pmatrix} \cos\theta_j & \sin\theta_j \\ -\sin\theta_j & \cos\theta_j \end{pmatrix}.$$

The matrix function $\mathbf{G}_j$ is then expressed as a function of the scalar variables: ω_j, which is the spatial frequency of the function; σ_j, the spatial extent of the function; (x_{0j}, y_{0j}), the origin of the function; and θ_j, the orientation of the function.

To perform the filtering operation the binary image is converted to a real valued image with a step height between levels of one and a zero mean value, $\mathbf{I}$. Since the set of Gabor functions is incomplete and non- orthogonal, the filtering must be performed by least squares optimization. On the small images discussed here direct methods are more efficient for this operation than the neural net method proposed for data compression [4]. Given n different $\mathbf{G}_j$'s the filtering operation is based on obtaining a least squares fit to the image $\mathbf{I}$ by forming the matrix $\mathbf{A}$, each component of which is the inner product of the form:

$$a_{ij} = \mathbf{G}_i \cdot \mathbf{G}_j,$$

and the vector,

$$b_i = \mathbf{I} \cdot \mathbf{G}_i$$

and solving

$$\mathbf{b} = \mathbf{c}\mathbf{A}$$

for the optional filter coefficients, $\mathbf{c}$. Since the matrix $\mathbf{A}$ is the same for any n Gabor functions, the matrix is factored once, and only generation of $\mathbf{b}$ and back substitution of the factored **A** matrix is required to obtain each **c**. The image is converted to binary by forming:

$$\mathbf{I}' = \sum_{j}^{n} c_j \mathbf{G}_j$$

and thresholding $\mathbf{I}'$ at the mean value of the reconstructed image.

3. ART-1 Learning

The ART-1 algorithm developed by Carpenter and Grosburg [5] is ideally suited for self-organization of unconstrained fonts or hand printed characters. The calculations involved are well adapted to parallel calculations on a single bit processor and are naturally parallel across the image field.

3.1. Parallel Weight Selection

The ART-1 algorithm consist of finding two sets of weights, **Z**'s, over each of j active memory locations which have an optimal resonance with one of the i input images. The input and output weights $\mathbf{Z}_{up,j}$ and $\mathbf{Z}_{down,j}$, for the j active memory locations are updated for each of i images, $\mathbf{I}_i$, by calculating:

$$\mu_{i,j} = \mathbf{I}_i \cdot \mathbf{Z}_{up,j}$$

$$\bar{x}_i = \mathbf{I}_i \cdot \mathbf{I}_i$$

$$\bar{T}_{i,j} = \mathbf{I}_i \cdot \mathbf{Z}_{down,j} / \bar{x}_i$$

and finding the maximum μ for which $\bar{T} > \rho$, the sensitivity of the required resonance. When an image meeting the sensitivity constraint is found the appropriate weights are adapted using:

$$\mathbf{Z}_{up,j} = \mathbf{I}_i \cdot \mathbf{Z}_{down,j} / (0.5 + \bar{T}_{i,j})$$

and

$$\mathbf{Z}_{down,j} = \mathbf{I}_i \cdot \mathbf{Z}_{down,j}.$$

If no resonance is achieved for a given image, a new memory location is added to the active list and the weights of this "blank" location are adapted to the image. This process is continued with each image until all available memory locations are set or all of the input images are used. After all vacant memory location are used, each memory location is compared to the product $\mu_{i,j}\bar{T}_{i,j}$ and the memory location for which this product is largest is then updated.

3.2. Evaluation of Self-Organization

The evaluation of the self-organized classes is achieved by accumulation of statistics in a classification variable:

$$Z_{class,k,j} = Z_{class,k,j} + 1 \;\text{ if }\; \text{class of}(\mathbf{I}_i) = k.$$

This table can then be used to determine the maximum selection strength of each memory location for all images and to assign classes to images based on resonance performance achieved over all memory location and class assignments. This allows a new set of images to be learned wholly by example and divided into resonance classes based on the recognition results achieved in the training set.

4. Recognition

Recognition is achieved by finding the maximum strength of resonance for the weighted classes, $\max(w_{k,j})$, using:

$$w_{k,j} = \sum_{i=k} Z_{class,k,j} \times \mu_{i,j} \bar{T}_{i,j}.$$

In addition, the average resonance strength of the strongest weighted resonance,

$$P = \max(w_{k,j}/n_{w,k,j}),$$

where $n_{w,k,j}$ is the number of terms used to form each $w_{k,j}$, provides a confidence limit for the evaluation of classification errors. If the value of P is less than the confidence expected for correctly classified data in the training set, then items should be classified as unknown. This significantly reduces the undectable error rate.

5. Results

The methods described in sections 2, 3, and 4 were tested by constructing a recognition system for machine and hand printed digits. A typical test sequence consists of loading the images of the test data into the array processor and performing filtering, feature extraction, and classification on these images. Each test set is then divided into two 512 character samples. The first 512 characters are used for the construction of the ART-1 weights and the Z_{class} statistical array as discussed in section 3. The second 512 image sample is then used as described in section 4 to test the classification of previously unseen images for maximum resonance. All error rates shown are for the second 512 image test sample.

5.1. Machine Print Data

The test sample consisted of 1024 machine printed digits taken from a single set of laser printer output. The primary source of variation in the test sample can be traced to variation in threshold during scanning and segmentation. The digits were not centered in the field or scaled to fit the 32 by 32 image size used in feature extraction. Table 5.1 shows the number of errors, out of 512 possible, for unfiltered machine print. Both errors which would be found by imposing confidence limits and labeled "Unknown", and undetected errors labled "Wrong" are shown. When high sensitivity matching is used in the feature extraction phase, the undetected error rate is less than 1% and the detected error rate is about 6%. Table 5.2 shows the error rate for the same 1024 digits when optimized Gabor filtering is used prior to feature extraction and classification. The filter tends to enlarge and shape the digit and decreases the sensitivity of the image to edge position. Even at matching sensitivity of 0.75 the error rate is substantially reduced. Using high sensitivity matching no errors occur in a 512 digit sample. Typical classification time is 3ms.

5.2. Hand Print Data

Table 5.3 shows the classification error for a sample of 512 hand printed digits. The digits were taken from hand printed digits contained in the NIST hand print data base. Data from 50 different individuals was used in the test. The 25 individuals used in the learning phase were different from the 25 individuals used to test classification. Undetected error rates are about 6% and detected error rates are about 13%. Typical classification time is 44ms.

6. Conclusions

An ART-1 based method of feature extraction and classification has been developed on a parallel computer which provides very high accuracy rates for machine print and reasonable accuracy rates for hand printed data. The high accuracy rate for machine printed data is achieved through the use of optimal Gabor function based linear filters. Since the method used is totally unsupervised, any font or symbol set can be learned without altering the algorithm. The method degrades gradually in the presence of noise and performs well even at 20% noise levels. These features make the combination of optimal filtering and ART-1 feature detection and classification an attractive candidate for commercial character recognition systems.

References

[1] L. D. Jackel, H. P. Graf, W. Hubbard, J. S. Denker, D. Henderson, and Isabelle Guyon, "An Application of Neural Net Chips: Handwritten Digit Recognition", IEEE International Conference on Neural Networks, San Diego, Vol. II, pp. 107-115, 1988.

[2] A. Rajavelu, M. T. Musavi, and M. V. Shirvaikar, "A Neural Network Approach to Character Recognition", *Neural Networks*, **2**, pp. 387-393, 1989.

[3] P. M. Flanders, D. J. Hunt, S. F. Reddaway, and D. Parkinson, "Efficient high speed computing with the distributed array processor", *Proceedings of Symposium on High Speed Computer and Algorithm Organization*, U of Ill., pp. 113-128, 1977.

[4] J. G. Daugman, " Complete Discrete 2-D Gabor Transform by Neural Networks for Image Analysis and Compression", *IEEE Trans. on Acoustics, Speech, and Signal Processing*,**36**,pp. 1169-1179, 1988.

[5] G. A. Carpenter and S. Grossberg, "A Massively Parallel Architecture for a Self-Organizing Neural Pattern Recognition Machine",*Computer Vision, Graphics, and Image Processing*, **37**, pp 54-115, 1987.

[6] G. A. Carpenter and S. Grossberg, "ART 2: self-organization of stable category recognition codes for analog input patterns", *Applied Optics*, **26**, pp. 4919-4930, 1987.

Sensitivity	Wrong	Unknown
0.75	63	53
0.80	27	37
0.85	5	36
0.90	4	23
0.95	3	32

Table 5.1: Classification errors in a sample of 512 machine printed characters when no filter is used.

Sensitivity	Wrong	Unknown
0.75	23	48
0.80	21	15
0.85	0	0
0.90	0	0
0.95	0	0

Table 5.2: Classification errors in a sample of 512 machine printed characters when normalization and Gabor filtering are used.

Sensitivity	Wrong	Unknown
0.75	29	59
0.80	29	64
0.85	29	67
0.90	27	72
0.95	28	87

Table 5.3: Classification errors in a sample of 512 hand printed characters when no filter is used.