International Joint Conference on Neural Networks (IJCNN) Volume III
June 7 - 11, 1992, Baltimore, MD

Massively Parallel Neural Network Recognition

C L. Wilson, National Institute of Standards and Technology
wilson@magi.ncsl.nist.gov
Gaithersburg, MD 20899

Abstract

Neural networks have provided the potential for massively parallel implementation for some time, but system level vision applications have only recently been realized. This is because the requirements for a total vision system must include the capability for image isolation, segmentation, feature extraction as well as recognition. Two systems have been developed on a massively parallel array processor which will be used to illustrate the importance of these higher level functions.

The first system is a massively parallel character recognition system. The second system is a system for classification of fingerprints. Both of these systems demonstrate state-of-art accuracy but both need improvements to be commercially viable. The issue in the character recognition system is to provide this accuracy at a speed compatible with commercial requirements of 1 page/s. This will require more sophisticated higher level image parsing functions without loss of accuracy. The issue in fingerprint classification is the requirement for 99.7% accuracy at current speeds.

1 Introduction

This paper discusses two complete neural network recognition systems, a character recognition system and a fingerprint classification system. Both systems are implemented on a massively parallel, SIMD, computer system with 1024 processors arranged in a 32 by 32 grid [1]. This computer is capable of performing 10^{10} binary operations per second, has a 1.25G byte processor interconnection bandwidth, and performs floating point calculations with greater speed than a conventional super computer.

The first system is a massively parallel character recognition system. This system achieves character recognition decision accuracies of 95% with 10% rejects including all error sources. When isolated characters are used, recognition rates of 10,000 characters/second have been achieved and recognition accuracies of 98.9% with 10% rejection have also been achieved. This speed contrasts sharply with the integrated system speed of 30 seconds/page, recognizing 130 characters/page, or 4.33 characters/second for total systems recognition time. The recognition time, using neural networks, is 0.34% of the total time in this system. At the same time, the module which loads image data into the parallel processor uses 30% of the time and segmentation uses 58% of the total time.

The second system is a system for classification of fingerprints. The system uses ridge-valley direction to convert the image into alignment and classification features, alignment of fingerprint cores from ridge-valley directions as image alignment method, K-L transforms of ridge-valley directions as a feature extraction method and Multi-Layer Perceptrons, MLP, as a classification method. The ridge-valley direction detection takes 0.4 s/image, the alignment 0.1 s/image, the K-L transform takes 20 ms/image, and the classification takes 1ms/image. A classification accuracy of 93% is achieved with 10% rejects. The image processing prior to classification takes more than 99% of total processing time; the classification time is 0.03% of the total system's time.

Three neural network based methods are used in these systems for feature extraction, classification, and combined feature extraction and recognition. The K-L method is used [2] for feature extraction. This method is a self-organizing method [3] that maximizes the variance in a feature set by using the principal eigenfunctions of the covarience matrix of the feature set. In the fingerprint system, local ridge directions are extracted from the image and use in subsequent processing. In the character recognition system, character images are used directly in subsequent processing. A similar technique has also been used with wavelets for face recognition [4] and for Kanji character recognition [5]. This transform is dimension reducing. For characters, the 1024 bit image is converted to 32 features. For finger prints, 640 ridge valley direction x and y components are converted to 106 features.

The features generated by the K-L transform are used for training a MLP using Scaled Conjugate, SCG, Gradient optimization [6]. For the problems presented here, this method is from 10 to 100 times faster than backpropagation. Details of the method and procedures for obtaining the program are available in [6].

The FAUST method presents an alternative method of feature extraction and classification [7]. The FAUST architecture is one of several neural networks which provide self-organizing multi-map capabilities. The structure used is a multi-map procedure similar to those known to exist in the mid-level visual cortex [8]. As in previous work [9, 10, 11] the method must provide a parallel, multi-map, self-organizing, pattern classification procedure. This is achieved using a feed-forward architecture which allows multi-map features stored in weights acting as associative memories to be accessed in parallel and to trigger a symmetrically controlled parallel learning process. A diagram of the FAUST system is shown in figure 1. This method allows features of different data type, such as binary image patterns and multi-bit statistical correlations, to be updated in parallel. This capability is provided by the parallel pattern association and relevance paths shown in figure 1 and by the existence of separate input modules for each path.

A pattern comparison method is used to form a centralized learning control which is contained in the symmetric triggering learning control block which gates data into the learning block on the right of figure 1. This combined architecture is described by the acronym FAUST (Feed-forward Association Using Symmetrical Triggering). The three essential features of FAUST shown in this figure are: 1) different feature classes use individual association rules in the pattern comparison blocks, 2) different feature classes use individual learning rules as illustrated by the pattern modification blocks, 3) all feature classes contribute symmetrically to learning as illustrated by the functional symmetry of the pattern and relevance paths. For graphic clarity, two feature classes are shown in figure 1 but the architecture is not restricted to any number or type of feature classes.

2 Systems Design

The basic structure of the character recognition and fingerprint classification systems is similar. Both systems have a loading phase which includes image decompression on the host serial computer, an alignment phase and feature registration phase, a feature extraction phase, and a recognition phase. Two variants of the character recognition system are discussed. One uses K-L features and a 48-64-10 MLP [12] trained by SCG [13]. The other uses a self-organizing method, FAUST. The fingerprint classification system uses a ridge-valley based feature isolation and alignment method, K-L feature extraction, and a 106-128-5 MLP trained by SCG for classification.

The function of all systems is similar. A raster scanned image is input to the system and ASCII classifications are returned. For the character recognition system, the input image is a binary image of a page containing 8,000,000 pixels. For the finger print system, the input image is a 512 by 512 8-bit gray level image. The character recognition system returns a page of character classifications for each page image. The fingerprint classification system returns an ASCII character representing one of five fingerprint classifications.

3 Character Recognition System

The massively parallel character recognition system developed by NIST (NIST Model Recognition System) [14] has achieved recognition accuracy which is compatible with many commercial requirements. On isolated characters, recognition rates of 10,000 characters/second have been achieved and recognition accuracy of 95% with no rejections and 98.8% with 10% rejections have also been achieved on handprinted digits from NIST Special Database 1 [15]. This speed contrasts sharply with the integrated system speed of 30 seconds/page recognizing 130 characters/page, 4.33 characters/second, for total systems recognition time. Table 1 explains the difference. The recognition time, item 6 in the table, is 0.34% of the total time in the K-L system. At the same time, the load module uses 30% of the time and segmentation uses 58% of the total time.

The distribution of timings is different when the self-organizing FAUST method is used. Data from a system's run using FAUST is shown in table 2. In this case, the recognition takes 49% of the total system time and loading and segmentation take most of the rest.

A simple model is proposed to account for these large differences in speed for the K-L based system. In the loading phase of the system, 8,000,000 pixels are being processed. In the final phase, 130 bytes are being processed, 1040 bits. The data volume decreases by a factor of 8,000. The algorithms in the recognition part of the system are much more complex than those in the early modules but the data volume is far smaller. Segmentation involves the processing of large data volumes while utilizing complex algorithms. This strongly suggests that improvements in recognition speed will have little effect on the commercial applicability of these systems while improvements in higher level processing are critical.

4 Fingerprint Classification System

The fingerprint classification system differs from the character recognition system in several important ways. First, fingerprints are natural objects which have classes designated by humans. These classes merge smoothly into each other. The five classes used are right loop, left loop, whorl, arch and tented arch [16]. The classification is made using a 512 by 512 gray level image. The difficulty of compression of these images is illustrated by comparing tables 1 and 3. The decompression and load time for an 8,000,000 pixel binary image is 8.9 seconds and the decompression and load time for a 256,000 pixel gray image is 3.2 seconds. This illustrates the difficulty of compressing gray level images in a lossless way.

After the fingerprint image is decompressed and loaded into the array processor, alignment features are extracted. A rule based alignment method is then used to center the print in the image field [17]. After the print has been centered in the image field, classification features are extracted. Both alignment features and classification features are extracted using local ridge slope data.

The classification features are passed through a K-L transform to extract maximum variance features in much the way described in [2]. This reduces the feature set from 640 to 106 features. These reduced features are used to train a MLP using SCG optimization. Typical hidden layer values are 64 to 128 nodes. The output layer contains 5 nodes, one for each class. Typical classification accuracy is 85% to 95%.

A complete reject rate versus accuracy curve is given in figure 2. At a reject rate of 10% the network accuracy is 93%. This is substantially below the 99.7% required for this application. Increased accuracy should be obtainable by increasing the size of the training set.

5 Conclusions

Two conclusions can be drawn from these complete vision systems. First, with feature compression methods such as K-L transforms, small networks which provide both high speed and good accuracy can be constructed. Second, for future systems to be reasonably efficient, the higher level image processing methods must be capable of much greater speed.

Acknowledgement

The author would like to acknowledge the members of the image recognition group research term who contributed to the work described here: J. L. Blue, G. T. Candela, M. D. Garris, P. J. Grother, S. Janet, C. I. Watson and R. A. Wilkinson.

References

[1] P. M. Flanders, R. L. Hellier, H. D. Jenkins, C. J. Pavelin, and S. Van Den Berghe. Efficient high-level programming on the AMT DAP. *IEEE Proceedings: Special Issue on Massively Parallel Computers*, 79(4):524–536, 4 1991.

[2] P. J. Grother. Karhunen Loève feature extraction for neural handwritten character recognition. In *Proceedings: Applications of Artificial Neural Networks III*. Orlando, SPIE, April 1992.

[3] R. Linsker. Self-organization in a perceptual network. *Computer*, 21:105–117, 1988.

[4] M. V. Wickerhauser. Fast approximate factor analysis. In *Proceedings October 1991*. SPIE, Washington University in St. Louis, Department of Mathematics, 1991.

[5] T. P. Vogl andm K. L. Blackwell, S. D. Hyman, G. S. Barbour, and D. L. Alkon. Classification of Japanese Kanji using principal component analysis as a preprocessor to an artificial neural network. In *International Joint Conference on Neural Networks*, volume 1, pages 233–238. IEEE and International Neural Network Society, 7 1991.

[6] J. L. Blue and P. J. Grother. Training feed forward nerworks using conjugate gradients. In *Proceedings Febuary 1992*. SPIE, National Institute of Standards and Technology Gaithersburg Mayland, 1992.

[7] C. L. Wilson. A new self-organizing neural network architecture for parallel multi-map pattern recignition - FAUST. *Progress in Neural Networks*, 4, 1992. to be published.

[8] A. Rojer and E. Schwatz. Multi-map model for pattern classification. *Neural Computation*, 1:104–115, 1989.

[9] 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. In *IEEE International Conference on Neural Networks, volume II*, pages 107–115, San Diego, 1988.

[10] A. Rajavelu, M. T. Musavi, and M. V. Shirvaikar. A neural network approach to character recognition. *Neural Networks*, 2:387–393, 1989.

[11] C. L. Wilson, R. A. Wilkinson, and M. D. Garris. Self-organizing neural network character recognition on a massively parallel computer. In *Proc. of the IJCNN, volume II*, pages 325–329, June 1990.

[12] D. E. Rumelhart, G. E. Hinton, and R. J. Williams. Learning internal representations by error propagation. In D. E. Rumelhart and J. L. McClelland, et al., editors, *Parallel Distributed Processing: Explorations in the Microstructure of Cognition. Volume 1: Foundations*, chapter 8, pages 318–362. MIT Press, Cambridge, 1988.

[13] M. F. Møller. A scaled conjugate gradient algorithm for fast supervised learning. *Neural Networks*, to be published.

[14] M. D. Garris, C. L. Wilson, J. L. Blue, G. T. Candela, P. Groither, S. Janet, and R. A. Wilkinson. Massivelly parallel implementation of character recignition systems. In *Proceedings Febuary 1992*. SPIE, National Institute of Standards and Technology Gaithersburg Mayland, 1992.

[15] C. L. Wilson and M. D. Garris. Handprinted character database. *National Institute of Standards and Technology*, Special Database 1, **HWDB**, April 18, 1990.

[16] *The Science of Fingerprints*. U. S. Department of Justice, Washington, DC, 1984.

[17] J. H. Wegstein. An automated fingerprint identification system. *National Institute of Standards and Technology*, NBS Special Publication 500-89, Febuary 1982.

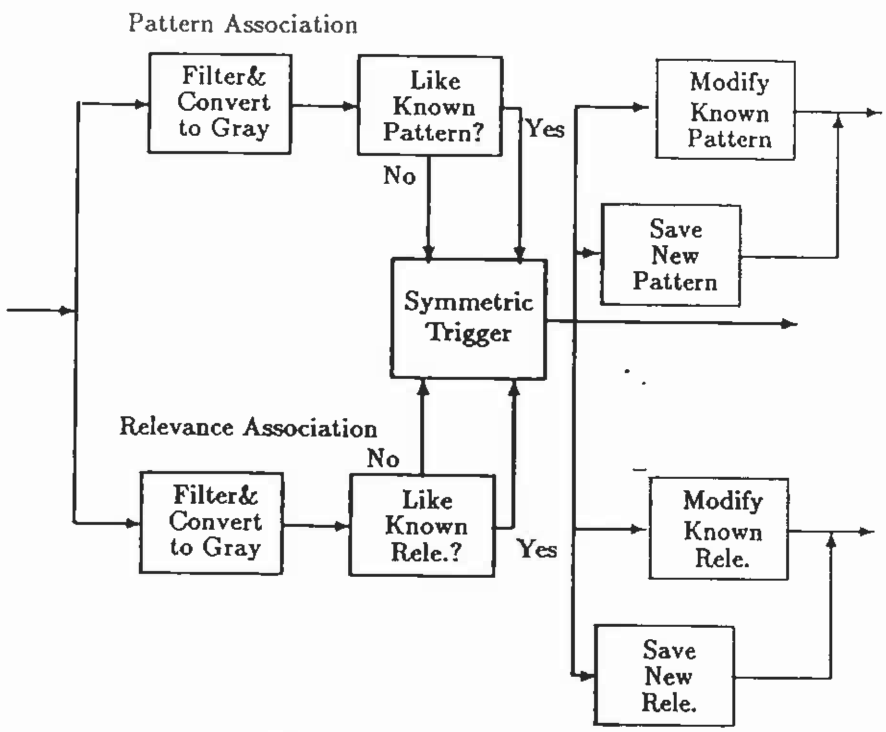

Figure 1: FAUST architecture disgram.

Load	8.912189	29.58
Isolate	1.746762	5.80
Segment	17.395340	57.74
Normalize	0.393859	1.31
Filter	1.242458	4.12
Recognize	0.102074	0.34
Reject	0.023852	0.08
Store	0.308159	1.02
Total	30.124693	100.00

Table 1: Module timing data for a K-L based character recognition system based on 2100 pages using 273,000 characters.

Load	9.262786	15.98
Isolate	1.747488	3.01
Segment	17.477033	30.15
Normalize	0.395426	0.68
Filter	0.026685	0.05
Recognize	28.653309	49.43
Reject	0.024382	0.04
Store	0.380781	0.66
Total	57.967891	100.00

Table 2: Module timing data for a FAUST based character recognition system based on 2100 pages using 273,000 characters.

Load	3.2	86.00
Alignment Features	0.2	5.38
Align	0.1	2.69
Classification Features	0.2	5.38
K-L Transform	0.02	0.54
Classify	0.001	0.03
Total	3.721	100.00

Table 3: Module timing data for a K-L based fingerprint classification system based on 1420 fingerprints.

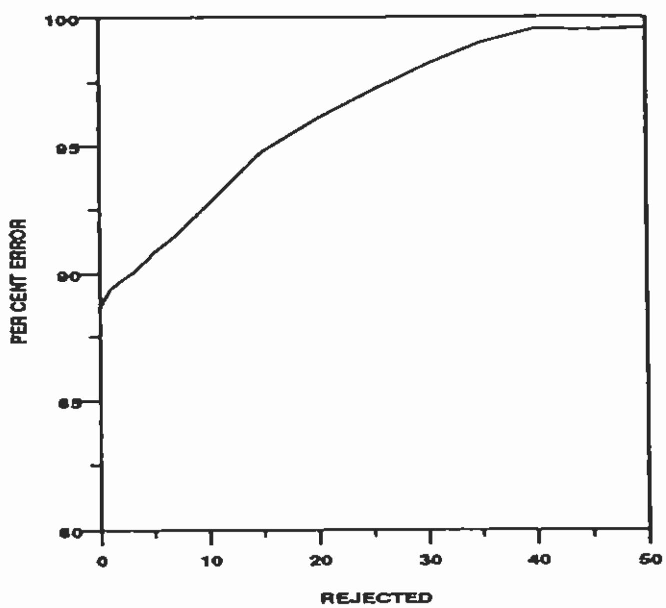

Figure 2: Reject versus accuracy curve for fingerprint classification.