A NEURAL APPROACH TO CONCURRENT CHARACTER SEGMENTATION AND RECOGNITION

Michael D. Garris and Charles L. Wilson
Computer Scientist and Mathematician
National Institute of Standards and Technology
Bldg. 225, Rm. A216
Gaithersburg, Maryland 20899

Presented at Southcon 92
Orlando, 1992

ABSTRACT

This paper presents a neural network solution that combines character segmentation and character recognition concurrently as a single task. Current segmentation methods utilize traditional image processing techniques such as spatial histograms which are only 60% accurate on handprint. Using traditional techniques for segmenting handprint in a model recognition system running on a massively parallel machine requires 55% of the entire processing time while the neural network classification requires 0.34% of the time. A neural network based solution for segmentation offers improvements in both speed and accuracy.

In order to demonstrate feasibility, initial experiments were conducted on machine printed digits. The results demonstrate that neural networks can be used for concurrent segmentation and recognition. Two different neural network solutions are studied, one based on a self-organized multi-map architecture, and the other based on the use of multi-layered perceptrons. Both approaches achieve 100% segmentation and recognition over a test set of 1,104 image samples. The multi-layered perceptron solution processes the activation signals from two separately trained networks whereas comparable results are achieved using the raw associations produced from a single self-organized network.

1. INTRODUCTION

Character recognition, the classification of well formed and cleanly segmented characters, has been studied in great detail in the past.[1-5] What is often avoided in character recognition research is the study of automated segmentation, the separation of text images into individual letters, one letter per image. Without this essential component, character-based classifiers are rendered useless.

A model recognition system has been implemented on a massively parallel computer at NIST.[6] The system consists of eight functional components. The loading of the image into the system and storing the recognition results from the system are I/O components. In between are components responsible for image processing and recognition. The first image processing component is responsible for image correction for scale and rotation, data field isolation, and character data location within each field; the second performs character segmentation; and the third does character normalization. Three recognition components are responsible for feature extraction and character reconstruction, neural network-based character recognition, and low-confidence classification rejection. Studies have shown that traditional image processing techniques used for character segmentation, even when implemented on a parallel computer, require 55% of the system's processing time at a rate less than 8 character/second.[7] A form containing 130 handprinted characters requires 17 seconds of processing just for character segmentation. This is much longer than the 1 second per page throughput required by many automated document processing applications. In order to improve segmentation, alternative methods are being explored.

One likely way of improving character segmentation throughput is via a neural network implementation. The model system's recognition component is implemented using a neural network which requires only 0.34% of the system's processing time. Using the parallel computer, classification rates as high as 10,100 characters/second can be realized. A neural network successfully trained to segment characters could be expected to provide similar performance.

Neural network-based segmentation has been the focus of recent work.[8,9] One interesting approach is to combine the tasks of character segmentation and character recognition into a single neural network solution. This also has been the focus of recent research.[10,11] The work presented in this paper is unique for several different reasons. First, the network solutions are small and robust; second, an unique class of objects is used for training; and third, the network solutions are implemented on a massively parallel computer.

2. ANTI-OBJECTS

Segmentation can be thought of as a two-object classifier. If a character segmentor is given a visual receptor field, implemented as a sliding window, then the task of the segmentor is to distinguish occurrences of a complete character in its visual field from occurrences of parts of neighboring characters. Images of isolated whole characters are referred to as *true objects*. The opposite of true objects are images centered on the space between two neighboring characters, referred to as *anti-objects*. The images contained in the segmentor's visual field will be referred to as frames. Figure 1 shows an example of a frame containing a true object on the left and a frame containing an anti-object on the right.

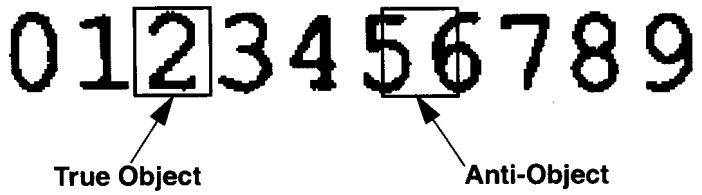

Figure 1. Example of a frame containing a true object (left) and a frame containing an anti-object (right).

A simple segmentor can be implemented by training a neural network to distinguish true objects from anti-objects. A window can be incrementally moved across a line of text producing a sequence of image frames. Frames classified as anti-objects demarcate character boundaries, and frames classified as true objects represent detected character images.

3. NETWORK ARCHITECTURES

Results from two different network architectures are studied in this paper. The first network presented is a self-organized multi-map algorithm developed at NIST. The second network solution studied uses a multi-layered perceptron (MLP) architecture.

3.1 Self-organized Network

The self-organizing multi-map algorithm is named Feedforward Association Using Symmetrical Triggering (FAUST).[12] FAUST provides a parallel, multi-map, self-organizing, pattern classification procedure similar to those known to exist in the mid-level visual cortex.[13] This neural network uses 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.

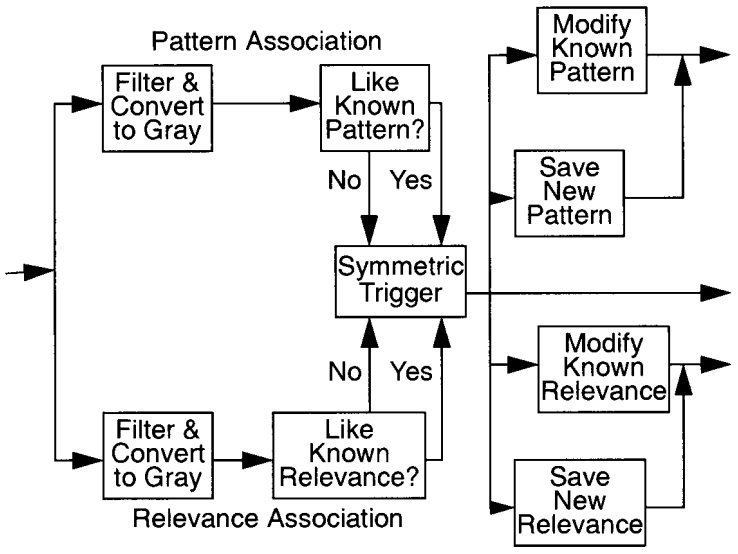

Figure 2. Diagram of the FAUST architecture.

A diagram of the FAUST architecture is shown in Figure 2. This network 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 2 and by the existence of separate input modules for each path. The number of feature types is shown as two (pattern and relevance) in the network diagram for this application, but the architecture is not restricted to any number or kind of feature types.

The FAUST network used in these experiments used two associative memories for each self-organized class. One is a pattern memory, an image that becomes an average of all images matched to the class. The other is a relevance memory, a memory that stores the importance of each pixel location in the pattern memory. No feature extraction is done; 1024-pixel binary images are the inputs to the network.

In the comparison units, the image is compared in parallel to the memories stored for each of the classes, and similarity values are computed. Two similarity values are computed for each class, one from the pattern comparison and one from the relevance comparison. The triggering unit decides if the image belongs to a known class or should be assigned to a new class. If both similarity values are above a given threshold, the image is assigned to the tentative class; the pattern memory and relevance memory use the image for learning. Otherwise a new class is begun and the image is used for learning by the new class. The total number of classes is determined by the training data and the threshold values chosen. After learning is completed on all images, each class is assigned to a character. Up to this time, the identity of each input image has not been used.

The three essential features of FAUST are: 1) Different feature classes use individual association rules for pattern comparison. 2) Different feature classes use individual learning rules for pattern modification. 3) All feature classes contribute symmetrically to learning.

3.2 MLP Network

Results using a MLP network, a more traditional neural network architecture, are also presented.[14] This network classifies by generating feedforward activations across a fully connected network containing an input layer, one hidden layer, and an output layer. Supervised training is done using Scaled Conjugate Gradient (SCG)[15] learning. Using the MLP architecture trained with Gabor feature vectors, character recognition accuracy of 99.8% for medium quality machine print has been demonstrated.[16]

Gabor functions are a set of incomplete nonlinear functions which reduce random image noise and smooth irregularities in image structure by acting as spatially localized low-pass filters. Gabor functions provide the minimum combination of uncertainty in position and spatial frequency resolution, and they match the visual receptor field profiles of mammalian eyes.[17] Figure 3 illustrates how Gabor functions can be tuned according to spatial extent, orientation, and phase.

These functions can be used in two different ways. Gabor reconstructed characters are enhanced by emphasizing the body of the character, reducing both the variations along its edges due to digitization and by normalizing its stroke width.[16] These functions can also be used to create feature vectors.

Each basis function applied separately to the character image produces a coefficient value. A feature vector of coefficient values can be computed by applying a set of basis functions to a single character image. This feature vector can then be used in place of the original character image by a neural network.

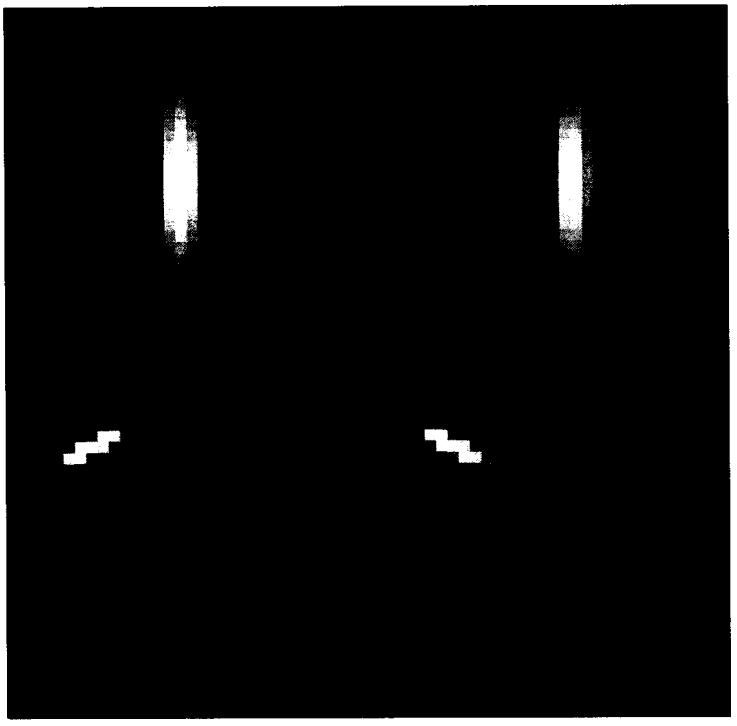

Figure 3. Example of 4 differently tuned Gabor functions.

The coefficient value vectors are especially useful in reducing the input dimensionality to MLP networks. For example, 32 tuned Gabor functions, when individually applied to an image, produce a vector of 32 coefficient values. The MLP networks studied in this paper were trained on these 32 coefficients rather than on the 1,024 pixels from the original character image. This dimensional reduction increases the generalization capabilities of the network[18], decreases classification times, reduces training times, and requires smaller sized training sets.

4. COMBINED SEGMENTATION AND RECOGNITION

Based on the theoretical segmentor described in Section 2, experiments were designed to combine segmentation with recognition. It was decided that initial experiments should be run on the relatively constrained environment of machine printed digits. Numerous experiments were conducted, two of which are discussed here.

4.1 Single Network Solution

The first set of experiments used a single trained network to perform both character segmentation and recognition. Networks were trained to classify the ten digit classes, "0" through "9", plus an additional class, the space character, which is simply an empty frame. Using this training strategy, the space character represents the most extreme case of anti-object, one which is void of any character image data.

4.1.1 FAUST Results

The FAUST network was trained using 1,100 input patterns. Each pattern was a character image scale normalized to 32 by 32 pixels. The training set contained 100 examples of each digit class, "0" through "9". In addition 100 space characters, blank white images, were included. The training digits were created by segmenting a full page of 12-point Courier machine print produced from a laser printer. The digits were scanned at 300 dots per inch binary, and a segmentor, using traditional image processing implemented on a massively parallel computer, was used to automatically segment the page of text.[7] Upon training, the self-organizing network created twelve pairs of memories, two classes for the digit "0", one class for each of the remaining nine digits, and one class for the space character.

A testing set was created by incrementally moving a window along a line of text and cutting successive frames. A line of 12-point Courier text containing 80 digits in the repeated order of "0" through "9" was used. The repeated order minimized the number of anti-objects used in these experiments. A window, 32 pixels in width, was incremented in steps of 2 pixels across the line of digits creating a sequence of 1,104 individual frames. A sequence of frames from the test set with a window increment of 4 pixels is shown in Figure 4.

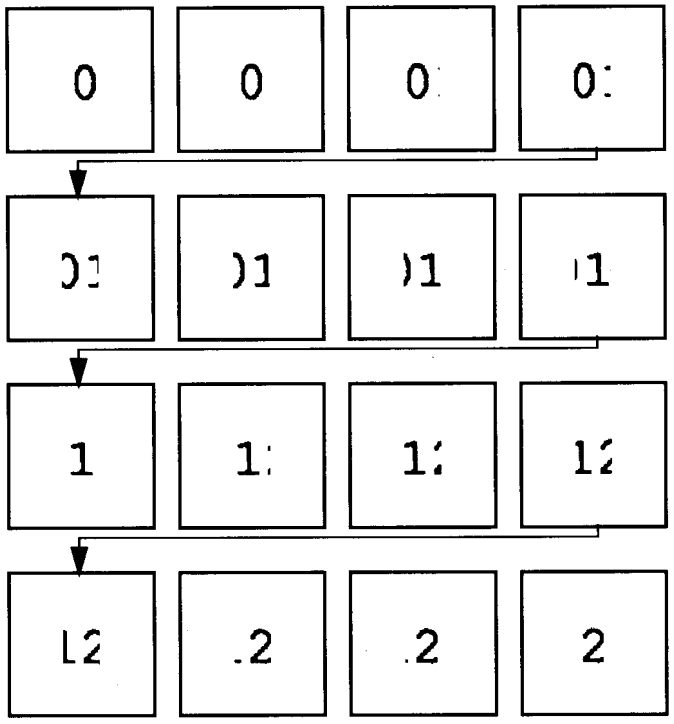

Figure 4. Sequence of test frames.

Figure 5 plots the associative memory activations produced when the trained FAUST network was presented the entire sequence of 1,104 testing frames. The signal responses produced from each frame presentation are plotted for the twelve self-organized memories. Successive frame presentations are represented horizontally, left to right. The associative memory responses are stacked vertically as separate signal channels. A repeating pattern in signal spikes can clearly be seen in this plot. As the frames proceed left to right, the true objects, "0"

through "9", periodically become centered. When this occurs, the appropriate FAUST memory fires strongly. Any time a frame contains an anti-object the network activation is absorbed by the memory associated with the space character (sp). The top associative memory in Figure 5 never receives any signal because it has a very low statistical relevance map due to its pattern memory being matched with only a single example of a "0". The results in Figure 5 demonstrates how a simple network trained on ten digit classes and a space character can be used to distinguish true objects from anti-objects and concurrently be used to classify the true-objects.

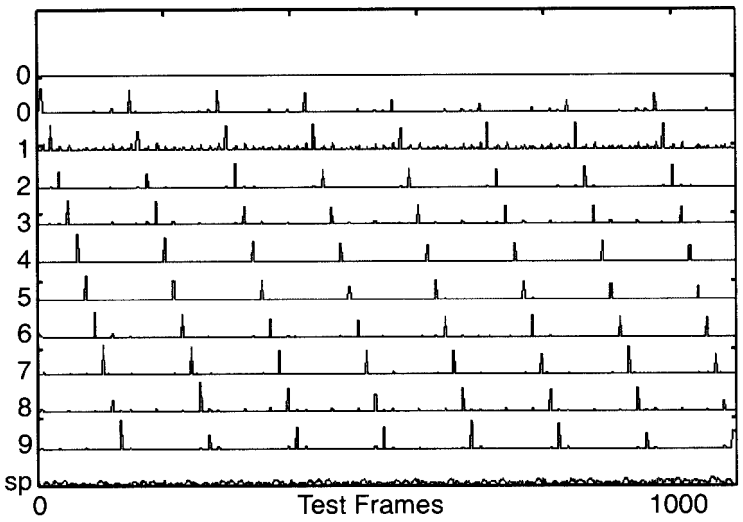

Figure 5. FAUST network responses.

4.1.2 MLP Results

Using the same single network strategy, a MLP network was trained using Gabor feature vectors. Each feature vector contained 32 coefficients created by applying 32 spatially tuned Gabor functions to a 32 by 32 scale normalized character image. The Gabor functions were tuned to the four quadrants of the normalized image with four equally spaced orientations within each quadrant and both sine and cosine phases.[15] The consistent dimensions of 32 are no coincidence. The massively parallel computer used to compute the coefficients contains two 32 by 32 grids of processing elements.

The MLP network was trained using 5,192 Gabor feature vectors. The training set contained 720 feature vectors from images of each digit class, "1" through "9", and 720 feature vectors representing the space character. The network contained 32 input neurodes, 32 hidden neurodes, and 11 output neurodes. After SCG training, the MLP network was tested using feature vectors computed from the same sequence of 1,104 frames used to test the FAUST network.

The results from testing the MLP network are plotted in Figure 6. The presentation of Gabor feature vectors computed from successive test frames is represented horizontally, left to right. The signals of the eleven output neurodes are stacked vertically. The results shown here are dramatically different from the results produced by the FAUST network in Figure 5. The output neurode activations of the ten digit classes contain significant noise.

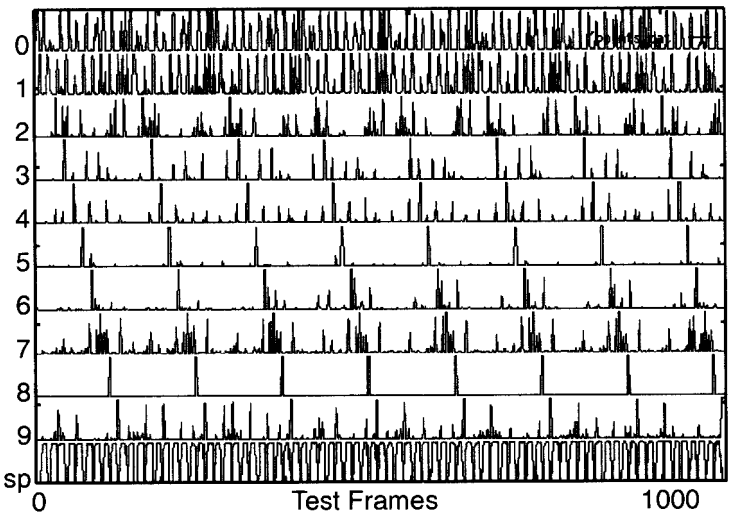

Figure 6. MLP network responses trained on 0-9 and space.

Notice that the signal plotted for the output neurode representing the space character is very periodic with local minima occurring at regular intervals. Upon inspection, it was found that the frames coinciding with the points of local minima contained images of centered characters. Figure 7 shows the results of using the signal from the space character as a shunt across the other channels so that when the space character signal is strong the network classifications are suppressed. The 10 signals in this figure are the result of multiplying the signals from the digit class neurodes in Figure 6 by one minus the signal from the space character neurode. Combining the MLP network's activations in this way removes the majority of noise shown in Figure 6. The same repeating pattern seen with the FAUST results in Figure 5 can be seen in Figure 7 with noise remaining within the "1" and "2" channels.

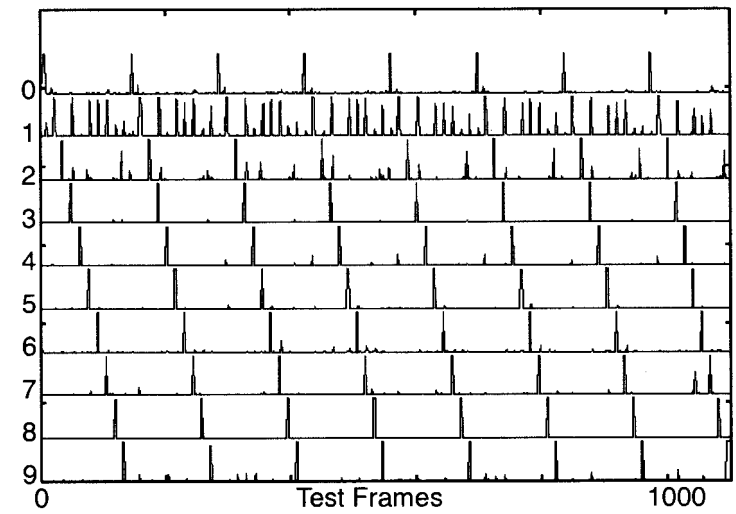

Figure 7. Signals from shunting 0-9 with the space signal.

The single network strategy, which works well with the FAUST network, is not as effective with the MLP network. This difference can be intuitively linked to the use of Gabor feature vectors. The spatially tuned Gabor functions used to produce the 32-coefficient feature vectors work as coarse stroke detectors.

Ambiguities arise when frames of anti-objects contain vertical strokes and curvature pieces of neighboring characters. These artifacts contribute to the false positive spikes shown in the signals for digit classes "1" and "2".

4.2 Two Network Solution

Experiments were conducted to improve the performance of the MLP network shown in Figure 7. A second MLP network was trained to classify frames into the two general categories, true objects and anti-objects. This new network is referred to as the object/anti-object network or OA network. Unlike the training of the single network solution, which generalized all anti-objects into a single space character, the OA network was trained with frames containing pieces of two neighboring characters with their intervening space centered in the frame.

The OA network was trained using 915 Gabor feature vectors. The training set contained 46 feature vectors from images of each digit, "1" through "9", assigned to the true object class. In addition, the training set contained 455 feature vectors computed from real anti-objects. Each anti-object was created by centering a window over the cut-point made by the page segmentor when separating two neighboring digits. The OA network was an MLP network containing 32 input neurodes, 16 hidden neurodes, and 2 output neurodes. The OA network was trained using SCG with all feature vectors of digits mapped to the first output neurode and with all feature vectors of anti-objects mapped to the second output neurode. In order to emphasize training on the space between characters, all 915 feature vectors were computed from images in reverse video, white print on a black background. The importance of reverse video training was not explored and is left to future study.

The OA network was tested with the same test set used with the single network solution, except in reverse video. Analyzing the activations of the OA network revealed a very well defined anti-correlation between the true object and anti-object neurodes. Based on this observation, a two network solution was designed. Figure 8 combines the shunted digit signals shown in Figure 7 with the true object signal (to) from the OA network.

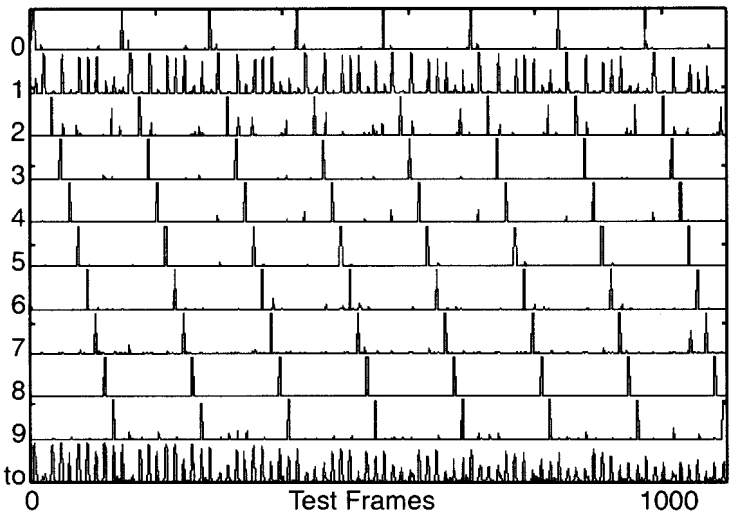

Figure 8. Shunted signals from Figure 6 with the true object signal from the OA network.

Figure 9 shows the results after combining the two network responses. This was accomplished by thresholding the OA network's true object signal and creating a logical mask at points of local maxima. All digit signals from the first network not coinciding with local maxima in the true object signal from the OA network were set to 0. This is equivalent to only accepting classifications from the first network's shunted digit signals when the OA network's true object activation is significantly high. This two network solution results in 100% correct segmentation and recognition when presented the 1,104 frame testing set.

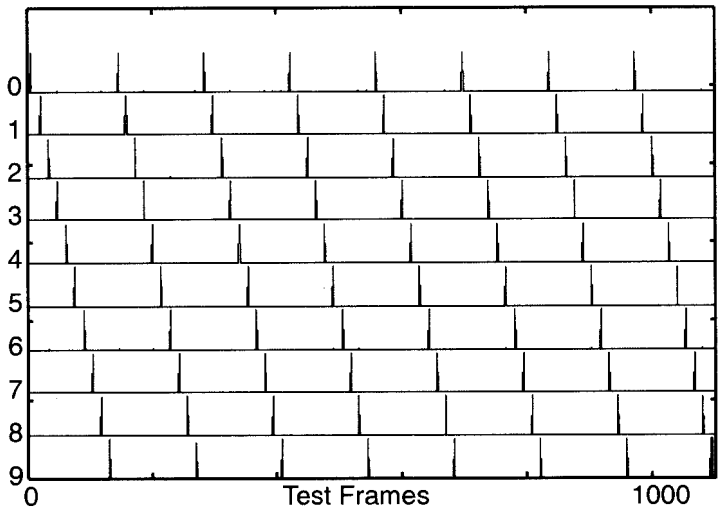

Figure 9. Results produced by combining the signals from two MLP networks, one trained to classify 0-9 and space, the other trained to distinguish true objects from anti-objects.

5. MERITS AND LIMITATIONS

The merits of the work in this paper are three-fold. First, the network solutions presented are based on the definition of anti-objects. This concept provides a new implementation technique whereby simple and yet powerful training strategies are possible. This was demonstrated through the single network solution based on the self-organizing FAUST algorithm. The FAUST network was successfully trained on a single space character which served as a generalization of all possible anti-objects. This generalization capability is important because for n true objects there are $O(n^2)$ anti-objects. Second, the MLP implementation using two separately trained networks is very compact, avoiding the failure of generalization which is associated with large networks.[18] Third, the anti-objects used in training the MLP networks did not require manual labeling of center points and were automatically derived from a segmentor based on traditional image processing techniques.[7] This greatly simplified the building of anti-object training sets.

The results presented in this paper have been based on machine printed digits. All training and testing has been conducted on a single font style and size. These solutions assume that a near-optimal window size and increment for creating frames exists. The implications of this assumption are small when working in the relatively constrained environment of machine print. However, it is likely that the details of this assumption will have to be addressed in order to successfully

apply these solutions to handprinted text. Dynamic window control will be the focus of future research.

6. CONCLUSIONS

Neural network solutions show great promise in improving the speed and accuracy of character segmentation. In addition, concurrent segmentation and recognition solutions are possible. Two solutions based on the definition of anti-objects have been studied. The first solution was based on a single network trained to classify machine printed digits and a space character. This strategy was very successful using the self-organized FAUST network. The same training strategy when applied to an MLP network trained with Gabor feature vectors contained a high number of false positive classifications. This can be attributed to ambiguities existing in the feature space. To eliminate these false positives, a second MLP network was trained to distinguish true objects from real anti-objects. Using the second network's activations as a mask, the false positives were completely eliminated. Both solutions resulted in 100% correct segmentation and recognition when presented a set of 1,104 test images.

REFERENCES

1. M. D. Garris, R. A. Wilkinson, and C. L. Wilson, "Methods for Enhancing Neural Network Handwritten Character Recognition," *International Joint Conference on Neural Networks*, Vol. I, pp. 695-700, Seattle, 1991.
2. G. E. Hinton, and C. K. I. Williams, "Adaptive Elastic Models for Character Recognition," *Advances in Neural Information Processing Systems*, R. Lippmann, Vol. IV, to be published, Denver, 1991.
3. L. D. Jackel, H. P. Graf, W. Hubbard, J. S. Densker, D. Henderson, and Isabelle Guyon, "An Application of Neural Net Chips: Handwritten Digit Recognition," *International Joint Conference on Neural Networks*, Vol. II, pp. 107-115, San Diego, 1988.
4. G. L. Martin and J. A. Pittman, "Recognizing Hand-Printed Letters and Digits," *Advances in Neural Information Processing Systems*, D. S. Touretzky, Vol. 2, 405-414, Morgan Kaufmann, Denver, 1989.
5. C. L. Wilson, R. A. Wilkinson, and M. D. Garris, "Self-Organizing Neural Network Character Recognition on a Massively Parallel Computer," *International Joint Conference on Neural Networks*, Vol. II, pp. 325-329, San Diego, 1988.
6. M. D. Garris, C. L. Wilson, J. L. Blue, G. T. Candela, P. Grother, S. Janet, and R. A. Wilkinson, "Massively Parallel Implementation of Character Recognition Systems," SPIE Machine Vision Applications in Character Recognition and Industrial Inspection, to be published, San Jose, 1992.
7. R. A. Wilkinson, "Segmenting Text Images with Massively Parallel Machines," SPIE Intelligent Robots and computer vision X, to be published, Boston, 1991.
8. K. Fukushima, T. Imagawa, and E. Ashida, "Character Recognition with Selective Attention," International Joint Conference on Neural Networks, Vol. I, pp. 593-598, Seattle, 1991.
9. H. P. Graf, C. Nohl, and J. Ben, "Image Segmentation with Networks of Variable Scale," *Advances in Neural Information Processing Systems*, R. Lippmann, Vol. IV, to be published, Denver, 1991.
10. J. D. Keeler and D. E. Rumelhart, "Self-Organizing Segmentation and Recognition Neural Network," *Advances in Neural Information Processing Systems*, R. Lippmann, Vol. IV, to be published, Denver, 1991.
11. G. L. Martin, "Centered-Object Integrated Segmentation and Recognition for Visual Character Recognition," *Advances in Neural Information Processing Systems*, R. Lippmann, Vol. IV, to be published, Denver, 1991.
12. C. L. Wilson, "A New Self-Organizing Neural Network Architecture for Parallel Multi-map Pattern Recognition - FAUST," *Progress in Neural Networks*, Vol. 4, to be published, 1992.
13. A. Rojer and E. Schwatz, "Multi-map Model for Pattern Classification," *Neural Computation*, 1:104-115, 1989.
14. D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning Internal Representations by Error Propagation," *Parallel Distributed Processing: Explorations in the Microstructure of Cognition*, D. E. Rumelhart, J. L. McClelland, et al., Volume 1: Foundations, pp. 318-362, MIT Press, Cambridge, 1986.
15. M. F. Moller, "A Scaled Conjugate Gradient Algorithm for Fast Supervised Learning," *Neural Networks*, Pergamon Press, New York, 1991.
16. M. D. Garris, R. A. Wilkinson, and C. L. Wilson, "*Analysis of a Biologically Motivated Neural Network for Character Recognition*," Proceedings: Analysis of Neural Network Applications, ACM Press, New York, 1991.
17. J. G. Daugman, "Complete Discrete 2-D Gabor Transform by Neural Networks for Image Analysis and Compression,", *IEEE Trans. on ASSP*, Vol. ASSP-36, pp. 1169-1179, 1988.
18. I. Guyon, V. N. Vapnik, B. E. Boser, L. Y. Bottou, and S. A. Solla, "Structural Risk Minimization for Character Recognition," *Advances in Neural Information Processing Systems*, R. Lippmann, Vol. IV, to be published, Denver, 1991.