Communicated by Alex Waibel

Speaker-Independent Digit Recognition Using a Neural Network with Time-Delayed Connections

K. P. Unnikrishnan*
Molecular Biophysics Research Department,
AT&T Bell Laboratories, Murray Hill, NJ 07974 USA

J. J. Hopfield
Molecular Biophysics Research Department,
AT&T Bell Laboratories, Murray Hill, NJ 07974 USA
and
Divisions of Chemistry and Biology, California Institute of Technology,
Pasadena, CA 91125 USA

D. W. Tank
Molecular Biophysics Research Department,
AT&T Bell Laboratories, Murray Hill, NJ 07974 USA

The capability of a small neural network to perform speaker-independent recognition of spoken digits in connected speech has been investigated. The network uses time delays to organize rapidly changing outputs of symbol detectors over the time scale of a word. The network is data driven and unclocked. To achieve useful accuracy in a speaker-independent setting, many new ideas and procedures were developed. These include improving the feature detectors, self-recognition of word ends, reduction in network size, and dividing speakers into natural classes. Quantitative experiments based on Texas Instruments (TI) digit data bases are described.

1 Introduction

Accurate recognition of spoken words in connected speech is difficult to achieve with limited computational resources. A "neural network" approach using time delays to organize the incoming signal into a recognizable form was constructed in earlier work, and studied in detail for

*Present address: Computer Science Department, GM Research Laboratories, Warren, MI 48090-9055 USA.

the case of a single speaker (Unnikrishnan *et al.* 1988, 1991). The case of a single speaker is, however, notoriously easier than speaker-independent word recognition, and is of rather limited utility in the world of engineering compared to the case of speaker independence. The present paper studies time-delay networks for speaker-independent connected speech.

The problem of identifying the spoken digits 0–9 in connected speech was chosen because it is well defined, small enough to study in detail, and has an established data base used as a standard for intercomparisons of results (Leonard 1984). This data base is sufficiently diverse that adequate performance on it is believed to be sufficient for field use in the United States. In addition, this particular problem is sufficiently important that a compact, low cost, and low power-consumption solution to it would be commercially useful.

The multiple-speaker problem is much more difficult than the single-speaker case, and its adequate solution demands many additional ideas and methods not present in our previous studies. Based on how well the original network performed on a speaker-dependent data base, we set out to examine whether a small number of networks could be used in parallel to solve the more difficult speaker-independent problem. Each subnetwork would be optimized on a separate cluster of data, for example, males or females. Because it is simple to train networks that make few mistakes of erroneous recognition, parallel use of multiple networks is a feasible approach to the general problem. In the course of these studies we found that even when the data were clustered into a few simpler problems, recognition accuracy was inadequate. Changes were therefore made to improve network performance. The most important of these changes are improved front-end signal processing for more reliable generation of invariant features from the input speech, reduction in the size of the network to favor generalization over memorization in the learning process, using the network itself to recognize what to learn, automatic segmentation of spoken digits from multiword strings, and explorations of dividing speakers into natural classes to simplify the problem faced by a single network.

In this paper we describe the performance of the various networks and approaches, presenting critical experiments for deciding to incorporate or abandon particular ideas and structures in the overall scheme. These results are described approximately in the order in which they were obtained. They begin with the obvious: using the same network that had proved successful for the single-speaker problem on the multiple-speaker data base. They conclude with experiments on a much-improved network and a data base of male speakers only (having found along the way that a single network shares with the simple hidden Markov model (HMM) performance at only a moderate level when men and women are placed together in the data base). The size and complexity of the networks simulated are such that an analog CMOS implementation would require less than a square centimeter.

2 Network Architecture and Learning Algorithm

The conceptual design of the word-recognition neural network is sketched in Figure 1. Details of the architecture and the learning algorithm have been described elsewhere (Unnikrishnan *et al.* 1991) and here we give only a very brief summary of it.

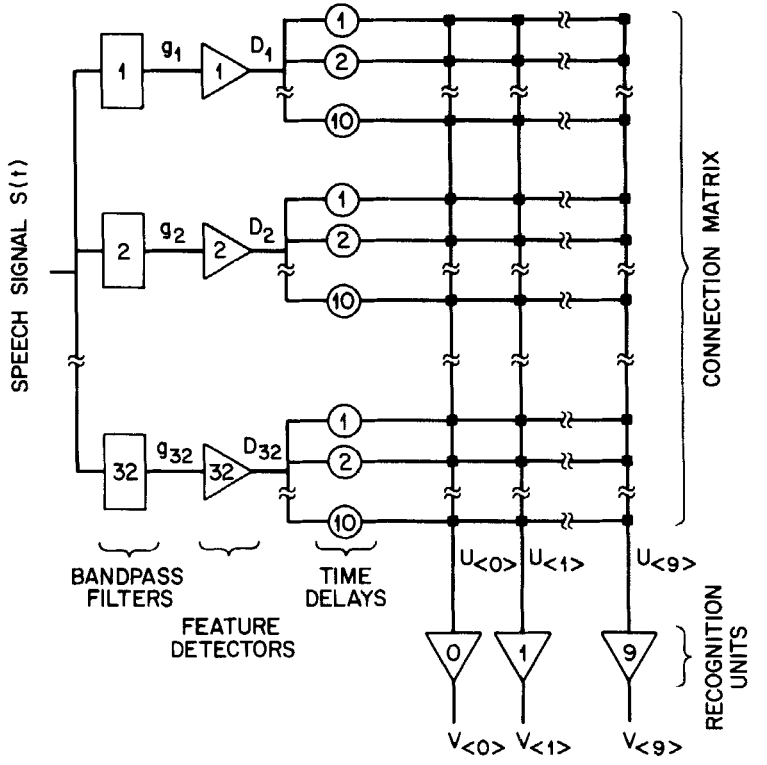

Figure 1: Block diagram of the speech recognition circuit with 32 bandpass filters. The feature detectors use a "center-surround" mechanism and the responses are generated by comparing outputs of neighboring filters. In some of the experiments, a different front end was used. It consisted of 15 bandpass filters and a zero crossing detector. The zero crossing detector uses raw speech waveform. The time delays used in the network have gaussian profiles with different widths. The connection matrix is learned using a learning algorithm and the rest of the circuit is fixed.

The analog speech waveform first goes to a bank of parallel analog frequency band filters. The rectified output from this filter bank is sent to a feature detector network that identifies the presence of short time scale features by a competitive comparison of the outputs of the filter bank. This procedure converts the original single-channel analog signal to multiple channel binary outputs crudely describing the frequency location of three significant peaks in the power spectrum. These outputs change much more slowly in time than the input signal does, and provide a suitable pattern to be recognized by neural network approaches. The multiple channel signal is sent down parallel tapped dispersive delay lines. The recognition unit for a particular word is connected by analog "weights" to these delay lines and the outputs of these nonlinear units signify recognition of the completion of particular words. The weights in the network are learned using an algorithm that minimizes a mutual discrimination error measure (see Hopfield 1987 and Unnikrishnan *et al.* 1991 for details of the learning algorithm).

By storing the delay information at discrete time intervals, the learning problem for a particular recognition unit can be reduced to a single layer of weights in an analog network with many inputs and a single output. The learning rule uses the analog response of the output units as a representation of the probability of a word having been said. For a given set of data, gradient descent leads to a unique set of weights for this learning rule and network structure.

To compensate for the temporal distortions, we have used dispersive time delays in the network (see also Tank and Hopfield 1987). These delays have gaussian profiles with different widths. In addition, each recognition unit also has an integration time constant. The summed signal from all delays is filtered with this time constant at the input of each recognition unit. Hence there are two parameters that determine the temporal tolerance of the network: (1) the width of the time delays (σ) and (2) the integration time constant of recognition units (τ_{rec}).

3 Data Base and Scoring Protocols

All the results reported here are based on two spoken digit data bases from TI. The TI isolated digit data base used consists of two utterances of each digit by 112 speakers, a regionally balanced mixture containing both men and women. These files were divided into a training set containing 50 speakers and a test set of 62 speakers. There is an appreciable variance in the distribution of utterance lengths. The average fractional time distortion [(longest − shortest)/average] for the training data was about 92%, but individual cases were as high as 157%.

The TI connected digit data base contains 330 speakers, a balanced mixture including children as well as men and women. There are two examples from each speaker of each individual digit, 11 examples from

each speaker of spoken digit pairs, and strings of up to 7 digits (Leonard 1984).

The experimental results for any particular set of data, network structure, and connections can be described by the percentage of correctly identified digits. Two measures were used to evaluate performance and provide recognition accuracy scores. The first is the threshold score: according to this measure, a recognition is scored as correct *only* if the output of the correct recognition unit was above a threshold value near the end of the utterance, with all the incorrect recognition units remaining below this threshold throughout the utterance. The second performance measure is the area score: according to this measure, a recognition is scored as correct if the time integrated output of the correct recognition unit over the period of the utterance is larger than the integrated output of any of the incorrect units. The threshold criterion for recognition is required for *word spotting* (recognition of individual words independent of the context in which they occur). The area criterion requires segmentation of data for recognition and is analogous to the scoring procedure used in HMM models (Levinson *et al.* 1983). We include recognition accuracies according to the area criterion for comparison with the results of other groups. Real time recognition of words will be impossible with the area criterion without cumbersome algorithmic additions to an otherwise simple network structure. In the models that use such a criterion, the recognition is usually done by waiting until the end of the sequence and then doing multiple matches with respect to the number of words spoken and possible candidates. The threshold criterion is a more strict measure of performance of the network than the area criterion and hence in all cases the threshold score is lower. In the following text and table we give the recognition accuracy with the area criterion outside the parentheses and the accuracy with the threshold criterion within parentheses.

4 Results

It was previously demonstrated (Unnikrishnan *et al.* 1991) that on a single-speaker connected-digit data base, a network with learned time-delayed connections had a performance similar to that provided by HMM digit recognition systems. When trained on 288 utterances of the digits, the network was able to learn all of the training data (recognition accuracy — 100% with threshold and area criteria). It could recognize a test set of 144 utterances with an accuracy of 100% (99.3% with threshold criterion).

To evaluate the extent to which this same network and training algorithm could solve the speaker-independent isolated digit recognition problem, it was trained on 500 utterances from the TI isolated digit data base. This data base contains a mixture of males, females, and children. The recognition accuracy on the training set was 98.6% [(91.4%

Table 1: Recognition Accuracy of Training and Testing Data with Various Network Configurations and Data Sets.[a]

Row	Recognition accuracy (%)		Comments
	Training data	Testing data	
a	98.6 (91.4)	81.5 (61.8)	32 input channels, isolated digit data base, 50-speaker training set, 62-speaker test set
b	99.8 (98.4)	92.0 (78.0)	As in case (a) but learning with self-justification
c	99.6 (96.1)		Learning on the combined sets of (a), 112 speakers
d	(98.5)		As in (c) but with 15 frequency channels and the "unvoiced" channel
e	97.6 (90.0)		Trained on 309 speakers, one example of each digit from each speaker
f	100 (99.5)	98.3 (92.6)	Train on one utterance of 110 males, test on other utterances of same males
g	99.8 (93.7)		Trained on 2090 one-word segments from two-digit strings of 110 males
h	99.9 (95.2)	95.6 (81.1)	Train on 1056 segments from two-digit strings of 55 males marked as training set, tested on 1037 segments from other 55 males
i	99.6 (93.5)	97.5 (84.9)	As in (h), but adding an additional 544 segments from three-digit strings of same speakers to training data; test data are same as in (h)
j	98.0 (83.3)	95.5 (75.4)	As in (i) but adding 1100 isolated word files of same speakers to training data
k	(92.6)	(82.4)	Recognition accuracy for male connected speech; trained on segments from connected speech set, tested on strings from the test set

[a]Recognition accuracy using area criteria is given outside the parentheses and the accuracy using threshold criteria is given within the parentheses. Rows a–d contain recognition results on the TI isolated-digit data base and rows e-k contain results on the TI connected-digit data base. All results in rows e-k are using a front end with 15 frequency channels and an "unvoiced" channel. See text for more details.

with threshold criterion); row a, Table 1]. It recognized an independent test set (different speakers) of 620 utterances with an accuracy of 81.5% [(61.8%); row a, Table 1]. These scores indicate that the circuitry and the learning paradigm as was used in the single speaker case was not sufficient for reliable recognition using the multiple-speaker data base.

4.1 Time-Duration Clustering. A series of experiments were done to determine the effects of temporal distortions on the network performance. In the first set of experiments, the data base was split into two clusters

(one containing the shorter utterances and the other containing the longer utterances) and separate networks trained on each one of them. In the next set of experiments, the time delays were made rigid. These did not change the network performance drastically, suggesting that for this data base, most of the difficulty may be due to variance in the frequency domain.

4.2 Frequency Clustering. Using a network trained on the entire data base, the files were split into two clusters: one containing high frequency utterances and the other containing low frequency utterances. Networks were able to learn the utterances in these clusters to better accuracies than those from an unbiased group taken from the same total data set. Also, a network trained on one cluster recognized test sets from the other cluster very poorly. These results demonstrate that spectral variance in acoustic features contribute substantially to the limited performance of the speech recognition network. We therefore adopted the premise that any complete recognition system would have two separate networks devoted to different frequency clusters and focused on improving the accuracy of a network for the male speaker data base subset.

4.3 Self-Justification. The speech examples were end-pointed by hand for use in the supervised learning procedure. But an analysis of the network outputs after learning showed that for many of the examples, the maximum response of the correct recognition unit was not at the assigned end point. This suggested that to generate optimal networks, the output of the network itself should be used for determining the end point of words. To accomplish this, the network was partially trained with the hand-justified end points. The time point of maximum response for the correct recognition unit, with these partially trained connections, was taken as the new end point and the training continued. This procedure implements self-justification of all the examples. This led to much better recognition of the training and testing data (compare row b with row a in Table 1), decreasing the error rate by about a factor of two on the independent test set.

4.4 Data Limitations. The TI isolated word multiple speaker data base was studied with the 32-channel front-end described in Unnikrishnan *et al.* (1991). Row b in Table 1 shows the results for training sets and test sets of approximately equal size. The excellent recognition score of (98.7%) according to the stringent threshold criterion on the training set was not matched by the score of (78%) on the test set. This discrepancy indicates that the system is to some extent memorizing the voices of the particular 50 speakers in the training set, and that the ensemble used for training is not large enough.

To examine whether the network is in principle capable of correctly classifying all the data at once, it was trained on all the data, comprising all 112 speakers and 1120 speech files. Row c shows the results. By the area criterion, the classification was near perfect, and with the threshold criterion, the performance was at the 96% level. This result suggested that to be able to both train and test the system, more speakers and more data per connection would be necessary. It also suggested that the network was near the end of its capability, and that some improvements would prove necessary to reach high levels of performance on an adequately broad data set.

The system requires too much data because it has too many connections to train. A typical word detector requires about $7 \times 32 + 1 = 225$ connections (on the average 7 time delays for each one of the 32 input channels and a bias value). While many more speech files than this are available, the similarity between different speakers or files is such that the amount of training data available is not adequately large. To alleviate this problem, we reduced the network size to 16 channels, with typically $7 \times 16 + 1 = 113$ connections per digit.

4.5 Zero-Crossing Detectors. The original 32 frequency band "front end" followed by a feature detector network was designed to locate peaks in the power spectrum and does not distinguish very well between vowel and consonant sounds or between voiced and unvoiced speech. A detector was designed to distinguish between voiced and unvoiced speech and used as one of the channels in a reduced 16 channel front end. A variety of methods can do this with high reliability. We chose a method based on zero crossings of the raw wave form as a method that would be easy to implement in analog hardware, and relatively independent of the intensity normalization of the sound.

An impulse was generated at the time of each upward zero crossing of the raw speech signal. These impulses were filtered with an integration time constant of 0.005 sec. The unvoicing channel was turned on if the output of this filter corresponded to a zero crossing rate which was above 2000 crossings per second, and if the total power in the speech was above a threshold level. The output of this channel located unvoiced consonants x, s, f, v, and t in the data set with excellent reliability. Further explorations in this paper have all been based on a 16-channel system containing the zero crossing detector. The other 15 channels are frequency channels of the previous type (see Unnikrishnan *et al.* 1991 for details), but having twice the frequency bandwidth. These channels were centered at the locations of the previous even-numbered channels 2–30. The feature detector network was modified slightly to prevent the identification of a peak in two adjacent frequency channels. The replacement of the 32-channel front end by the 16-channel system described above resulted in better performance on the entire 112 speaker data base (compare rows c and d in Table 1). Confronted with the necessity of

obtaining more data, and desiring to move toward connected speech, we began working with the much larger TI data base for connected speech.

4.6 Separating Males and Females. The TI connected-digit data base available to us contains a regionally balanced set of 309 speakers, including men, women, and children. When trained on one utterance of each speaker on each of the 10 digits spoken as an isolated word (3090 files), a relatively poor performance level [97.6% (90.0%); row e, Table 1] on the training set was achieved. Clearly the speech variation is now greater than the network can encompass. One major difference between the present data base and the previous one is the inclusion of children. Following the partitioning idea described earlier, we split the data base into two portions, males and nonmales.

For the male training set of isolated words from the connected digit data base (consisting of one example of each digit spoken by all 110 speakers) the network could be trained to a high level of performance on the training set [100% (99.5%); row f, Table 1]. The poorer performance on the test set from the same speakers (row f, Table 1) indicates that there is still an inadequate number of speech files in the training set. However, more data were now available from strings of two and three digits.

4.7 Automatic Segmentation of Training Data. To obtain individual words necessary for training from digit strings without a large amount of hand segmentation, a bootstrap procedure was employed. To begin, the recognition network with the connections learned from one utterance of each digit from the 110 male speakers was used to label the ends of words in the connected speech files. The recognition score was 100% [(99.5%); row f, Table 1] on the training set and 98.3% [(92.6%); row f, Table 1] on the test set. These connections were then used to segment individual digits from two-digit strings. The system could now be trained on this larger data base. By iteration, the total training set size was ultimately increased to 2090 utterances. This training set could be recognized with an accuracy of 99.8% [(93.7%); row g, Table 1]. Since the performance is lower by the threshold criteria than that obtained with the isolated digits database (row f, Table 1), we surmise that the recognition of segmented digits from strings is a harder problem than working with isolated words. The two obvious differences between this data base and the isolated word data base are the larger variation in the lengths of utterances and word–word coarticulation effects.

This enlarged data set was split into training and test sets (55 speakers for training and 55 speakers for testing) yielding 1056 segmented words for training and 1037 words for testing. The network could be trained to recognize the test set with an accuracy of 99.9% [(95.2%); row h, Table 1]. The test set was recognized with an accuracy of 95.6% [(81.1%); row h, Table 1]. The fact that training set could be learned very well and not

the test set shows that the total number of files in the training set is still small. We proceeded to increase the size of the training set by segmenting digits from three-digit strings.

Adding segments from three-digit strings yielded a total of 2600 training words. The network could be trained to recognize this training set with an accuracy of 99.6% [(93.5%); row i, Table 1] and to recognize the test data (same data set as in the previous case) with an accuracy of 97.5% [(84.9%); row i, Table 1]. But while the addition of new words to the training data increases the recognition accuracy for a test set, the continued poorer performance on the test set compared to the training set shows that there is still inadequate training data.

An experiment was tried in which the isolated digits were added to the segmented connected digits training data. The recognition score on the training set was reduced to 98% [(83.3%); row j, Table 1] and the score on test set was reduced to 95.5% [(75.4%); row j, Table 1]. The isolated digits typically have much longer duration than segmented digits from strings. The resultant additional variance in length is probably the cause of the reduced recognition accuracy.

4.8 Recognition of Strings. The experiments described above were done on isolated digits or digits segmented out from strings. We tested the performance of the network mentioned in row i of Table 1 on unsegmented connected digit strings from which the segmented test set had been previously produced. Some readjustment of delay-line parameters and integration time constants was necessary to eliminate the inhibitory signals from previous digits preventing recognition of a current digit. Such a network was able to recognize the training data with an threshold criteria accuracy of (92.4%) (row k, Table 1) and test data with a threshold accuracy of (82.4%) (row k, Table 1). We did not write the more complex software to do scoring for continuous speech by the area criterion, since this is not the desired ultimate use of the network. But by analogy with other experiments (cf, row i, Table 1) we would anticipate a recognition accuracy by the area criteria of approximately 99% on the training data set and 97% on the test data set.

More conventional approaches to this problem, involving extensive high-speed digital processing and a combinatorial examination of locations for word boundaries, have been carried out on this data base by many others. Using a network that uses acoustic-phonetic features, Bush and Kopec (1987) achieved an accuracy of 96%. Rabiner *et al.* (1988) achieved an accuracy of 97.1% using a hidden Markov model. Our network can easily be implemented using low precision and low power analog circuitry. The connections would require only two bits of precision and an algebraic sign and the network has been shown to tolerate a considerable amount of noise (Unnikrishnan *et al.* 1991). While the experiments are not strictly comparable (the earlier work is on string recognition, and we have not made a complete study of all strings of all

nonchildren), the difference between them is comparable to that expected between threshold and area criteria within our studies. This indicates that the two approaches are extracting similar amounts of information from the sound signal (though not necessarily the same information), and that the major addition of the HMM procedure is to be able to work somewhat better with words of great length variability through massive computation. The direct neural network approaches to this problem are to use multiple or hierarchical time-scale networks (see also Waibel *et al.* 1989).

5 Conclusions

We believe that the time-delay networks of this style we have studied are likely to be able to solve the speaker-independent digit recognition problem at a useful engineering level, with a neural network small enough to fit onto a single very low power analog VLSI chip. Even if four sets of connections (two sets of time delays and two sets of voice qualities) are needed, the total number of connections required is less than 6000. The four networks would share a common front end and delay network. The Intel 80170NW electrically trainable analog neural network chip based on eeprom technology already has 10,000 adjustable analog connections. The experiments we have described permit us to delineate the remaining problems and possible ways to solve them.

First, the front-end needs some improvement. The very large increase in performance produced by the inclusion of a voicing detector is an indication that the substitution of one or two frequency filters by more clever feature detectors would be of enormous help. Even the frequency filters themselves are not optimal. The output of the filter bank often lacks a formant trajectory when that trajectory is clearly visible in a windowed FFT power spectrum. The variability of our front-end output compared to that of the WAVES program (Entropic Speech Inc.) suggests that better filters alone would be of considerable help.

Second, the amount of available data in the TI data set is inadequate for the learning procedure used in the present study. It is, for example, responsible for the large difference in recognition accuracy between the test and the training set illustrated in row k of Table 1. A modified learning procedure that can capture outliers and generalizes better could be adopted, or alternatively a brute force approach of using a larger data set. For example, a variance model could be used in conjunction with the training set to effectively enlarge it.

Third, speaker clusters should be produced by the networks directly. In the experiments described here, training data were clustered using males and nonmales as predefined categories. This would make each cluster more compact and simplify the problem.

Fourth, when connected speech and isolated words are combined in a single data base, the difference in the duration of a given word within the data now begins to matter. This problem can be circumvented by dividing the data into fast and slow categories by clustering as illustrated in the text, and training a network for each cluster. These networks could be run in parallel, since false *recognitions* are generally not a problem. The output of the best network can then be used for recognition. The alternative use of a hierarchy of two delay time scales is also attractive.

Acknowledgments

The TI connected digit data base was provided by the National Bureau of Standards. We wish to thank David Talkin for providing us the WAVES program and the Speech Research Department at Bell Labs for computer support. The work of J. J. H. at Caltech was supported in part by Office of Naval Research (Contract No. N00014-87-K-0377).

References

Bush, M. A., and Kopec, G. E. 1987. Network-based connected digit recognition. *IEEE Trans. ASSP* **35**, 1401–1413.

Hopfield, J. J. 1987. Learning algorithms and probability distributions in feed-forward and feed-back networks. *Proc. Natl. Acad. Sci. U.S.A.* **84**, 8429–8433.

Leonard, G. E. 1984. A database for speaker-independent digit recognition. *Proc. Intl. Conf. Acoustics Speech Signal Process.* **3**, 42.11.1–42.11.4.

Levinson, S. E., Rabiner, L. R., and Sondhi, M. M. 1983. An introduction to the application of the theory of probabilistic functions of a Markov process to automatic speech recognition. *Bell System Tech. J.* **62**, 1035–1074.

Rabiner, L. R., Wilpon, J. G., and Soong, F. K. 1988. High performance connected digit recognition using hidden Markov models. *Proc. Intl. Conf. Acoustics Speech Signal Process.* **S3.6**, 119–122.

Tank, D. W., and Hopfield, J. J. 1987. Concentrating information in time: Analog neural networks with applications to speech recognition problems. *Proc. IEEE First Intl. Conf. Neural Networks*, San Diego, CA.

Unnikrishnan, K. P., Hopfield, J. J., and Tank, D. W. 1988. Learning Time-delayed connections in a speech recognition circuit. *Abstr. Neural Networks Comput. Conf.*, Snowbird, UT.

Unnikrishnan, K. P., Hopfield, J. J., and Tank, D. W. 1991. Connected-digit speaker-dependent speech recognition using a neural network with time-delayed connections. *IEEE Transact. Signal Proc.* **39**, 698–713.

Waibel, A., Hanazawa, T., Hinton, G., Shikano, K., and Lang, K. 1989. Phoneme recognition using time-delay neural networks. *IEEE Trans. ASSP* **37**, 328–339.

Received 11 February 1991; accepted 15 July 1991.