Spoken Letter Recognition

Ronald Cole, Mark Fanty

Department of Computer Science and Engineering
Oregon Graduate Institute of Science and Technology
19600 NW Von Neumann Dr.
Beaverton, OR 97006

Introduction

Automatic recognition of spoken letters is one of the most challenging tasks in the field of computer speech recognition. The difficulty of the task is due to the acoustic similarity of many of the letters. Accurate recognition requires the system to perform *fine phonetic distinctions*, such as B **vs.** D, B **vs.** P, D **vs.** T, T **vs.** G, C **vs.** Z, V **vs.** Z, M **vs.** N and J **vs.** K. The ability to perform fine phonetic distinctions—to discriminate among the minimal sound units of the language—is a fundamental unsolved problem in computer speech recognition.

We describe two systems that apply speech knowledge and neural network classification to speaker-independent recognition of spoken letters. The first system, called EAR (English Alphabet Recognizer), recognizes letters spoken in isolation. First choice recognition accuracy is 96% correct on 30 test speakers. A second system locates and recognizes letters spoken with brief pauses between them. First choice recognition accuracy is 95.7% on 10 test speakers for letters correctly located. This system was used to retrieve spelled names from a database of 50,000 common last names. Of the 68 names spelled by ten test speakers, 65 were retrieved as the first choice, and the remaining three were the second choice.

We attribute the high level of accuracy obtained by these systems to (a) accurate location of segment boundaries, which allows feature measurements to be computed in the most informative regions of the signal, (b) the use of speech knowledge to design feature measurement algorithms, and (c) the ability of neural network classifiers to model the variability in speech.

Isolated Letter Recognition

System Overview

Figure 1 shows the system modules that transform an input utterance into a classified letter. The system is able to accept microphone input or classify letters from digitized waveform files.

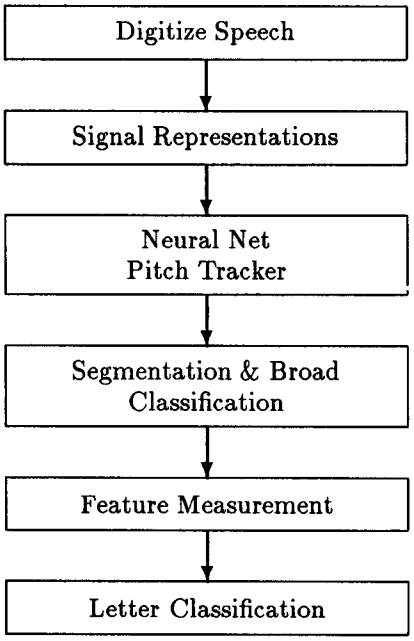

Figure 1: EAR Modules

Data Capture

Speech is recorded using a Sennheiser HMD 224 noise-canceling microphone, lowpass filtered at 7.6 kHz and sampled at 16 kHz. Data capture is performed using the AT&T DSP32 board installed in a Sun4/110. The utterance is recorded in a two second buffer using the WAVES+ software distributed by Entropic systems. In order to speed recognition time, the spoken letter—typically 400 to 500 msec long—is located within the 2 sec buffer based on values observed in two waveform parameters; the zero crossing rate and peak-to-peak amplitude. The remaining representations, such as the DFT, are then computed in the region of the utterance only.

Signal Processing

Signal processing routines produce the following set of representations. All parameters are computed every 3 msec.

zc0-8000: the number of zero crossings of the waveform in a 10 msec window;

ptp0-8000: the peak-to-peak amplitude (largest positive value minus largest negative value) in a 10 msec window in the waveform;

ptp0-700: the peak-to-peak amplitude in a 10 msec window in the waveform lowpass filtered at 700 Hz;

DFT: a 256 point DFT (128 real numbers) computed on a 10 msec Hanning window; and

spectral difference: the squared difference of the averaged spectra in adjacent 12 msec intervals.

Pitch Tracking

A neural network pitch tracker is used to locate pitch periods in the filtered (0-700 Hz) waveform [2]. The algorithm locates all plausible candidate peaks in the filtered waveform, computes a set of feature measurements in the region of the candidate peak, and uses a neural network (trained with backpropagation) to decide if the peak begins a pitch period. The neural classifier agrees with expert labelers about 98% of the time—just slightly less than they agree with each other.

Segmentation and Broad Classification

A rule-based segmenter, modified from [3], was designed to segment speech into contiguous intervals and to assign one of four broad category labels to each interval: CLOS (closure or background noise), SON (sonorant interval), FRIC (fricative) and STOP.

The segmenter uses cooperating knowledge sources to locate the broad category segments. Each knowledge source locates a different broad category segment by applying rules to the parameters described above. For example, the SON knowledge source uses information about pitch, ptp0-700, zero crossings and spectral difference to locate and assign boundaries to sonorant intervals.

Feature Measurement

A total of 617 features were computed for each utterance. Spectral coefficients account for 352 of the features. For convenience, the features are grouped into four categories, which are briefly summarized here:

- **Contour features** were designed to capture the broad phonetic category structure of English letters. The contour features describe the envelope of the zc0-8000, ptp0-700, ptp0-8000 and spectral difference parameters. Each contour is represented by 33 features spanning (a) the interval 200 msec before the sonorant; (b) the sonorant; and (c) the interval 200 msec after the sonorant. The 33 features are derived by dividing each of these intervals into 11 equal segments, and taking the average (zc0-8000, ptp0-700, ptp0-8000) or maximum (spectral difference) value of the parameter in each segment.
- **Sonorant features** were designed to discriminate among: (a) Letters with different vowels (e.g., E, A, O); (b) letters with the same vowel with important information in the sonorant interval (e.g., L, M, N); and (c) letters with redundant information near the sonorant onset (e.g., B, D, E). Sonorant features include averaged spectra in seven equal intervals within the sonorant, additional spectral slices after vowel onset (to determine the place of articulation of a preceding consonant), and estimates of pitch and duration.
- **Pre-sonorant features** were designed to discriminate among pre-vocalic consonants (e.g., V *vs.* Z, T *vs.* G) and to discriminate vowels with glottalized onsets from stops (e.g., E *vs.* B, A *vs.* K). These features include estimates of prevoicing, and spectra sampled within the STOP or FRIC preceding the SON. If no STOP or FRIC were found, features were computed on an interval 200 msec before the vowel.
- **Post-sonorant features** were designed to discriminate among F, S, X and H. Much of this information is captured by the contour features. The main post-sonorant feature is the spectrum at the point of maximum zero crossing rate within 200 msec after the sonorant.

Letter Classification

Letter classification is performed by fully connected feed–forward networks. The input to the first network consists of 617 feature values, normalized between 0 and 1. There are 52 hidden units in a single layer and 26 output units corresponding to the letters A through Z. The classification response of the first network is taken to be the neuron with the largest output response.

If the first classification response is within the E-set, a second classification is performed by a specialized network with 390 inputs (representing features from the consonant and consonant–vowel transition), 27 hidden units and 9 output units. Similarly, if the first classification response is M or N, a second classification is performed by a specialized network with 310 inputs (representing features mainly in the region of the vowel–nasal boundary), 16 hidden units and 2 output units. This strategy is possible because almost all E-set and M–N confusions are with other letters in the same set. If the classification response of the first network is not M or N or in the E-set, the output of the first net is final.

System Development

Development of the EAR system began in June 1989. The first speaker-independent recognition result, 86%, was obtained in September 1989. The system achieved 95% in January 1990 and 96% in May 1990. The rapid improvement to 95% in 5 months was obtained by improving the segmentation algorithm, the feature measurements and the classification strategy. The improve-

ment to 96% resulted from increased training data and the use of specialized nets for more difficult discriminations. This section briefly describes the research that lead to the current system.

Database

The system was trained and tested on the ISOLET database [4], which consists of two tokens of each letter produced by 150 American English speakers, 75 male and 75 female. The database was divided into 120 training and 30 test speakers. All experiments during system development were performed on subsets of the training data.

Segmenter Development

The behavior of the segmentation algorithm profoundly affects the performance of the entire system. Segment boundaries determine where in the signal the feature measurements are computed. The feature values used to train the letter classification network are therefore directly influenced by the segmenter.

The rule-based segmenter was originally developed to perform segmentation and broad phonetic classification of natural continuous speech. The algorithm was modified to produce optimum performance on spoken letters. It was improved by studying its performance on letters in the training set and modifying the rules to eliminate observed errors.

Feature Development

The selection of features was based on past experience developing isolated letter recognition systems [5] and knowledge gained by studying visual displays of letters in the training set. (Letters in the 30 speaker test set were never studied.) Several features were designed to discriminate among individual letter pairs, such as B and V. For these features, histograms of the feature values were examined, and different feature normalization strategies were tried in order to produce better separation of the feature distributions. Feature development was also guided by classification experiments. For example, a series of studies on classification of the letters by vowel category showed that the best results were obtained using spectra between 0-4 kHz averaged over seven equal intervals within the vowel.

Network Training

Neural networks were trained using backpropagation with conjugate gradient optimization [6]. Each network was trained on 80 iterations through the set of feature vectors. The trained network was then evaluated on a separate "cross–validation" test set (consisting of speakers not in the ISOLET database) to measure generalization. This process was continued through sets of 80 iterations until the network had converged; convergence was observed as a consistent decrease or leveling off of the classification percentage on the cross–validation data over successive sets of iterations. Convergence always occurred by 240 iterations, about 36 hours on a Sun 4/60.

The main (26 letter) network was trained with 240 feature vectors for each letter (6240 vectors), computed from two tokens of each letter produced by 60 male and 60 female speakers. The specialized E-set and MN networks were trained on the appropriate subset of letters from the same training set.

A	98.3	H	100.0	O	100.0	V	93.3
B	88.3	I	98.3	P	91.7	W	98.3
C	100.0	J	98.3	Q	100.0	X	98.3
D	93.3	K	96.7	R	100.0	Y	100.0
E	100.0	L	100.0	S	93.3	Z	96.7
F	96.7	M	88.1	T	90.0		
G	98.3	N	80.0	U	98.3		

Table 1: Classification performance for individual letters for 30 test speakers (with E-set and M–N nets).

Recognition Performance

The EAR system was evaluated on two tokens of each letter produced by 30 speakers. The main network (26 outputs, no specialized nets) performed at 95.9%. The specialized E-set network improved performance slightly, while the MN network hurt performance on this data set (experiments on subsets of the training data showed substantial improvement with the MN network). The combined three–network system performed at 96%. Table 1 shows the individual letter scores for the combined three–net system. The specialized E-set network scores 95% when run on all the E-set, and scores 94.2% when trained and tested on just B,D,E and V.

Multiple Letter Recognition

The approach used to classify letters spoken in isolation has been extended to automatic recognition of *multiple letters*—letters spoken with brief pauses between them. We have implemented and evaluated a system that uses multiple letter strings to retrieve names from a database of 50,000 common last names.

The recognition system differs from EAR in two important ways: (a) the DFT was reduced to 128 points, and (b) a neural network was used to segment speech into broad phonetic categories.[1] The processing stages are shown in Figures 2 and 3.

Neural Network Segmentation and Broad Classification

The neural network segmenter, developed by Murali Gopalakrishnan as part of his Master's research, con-

[1] Performance of the EAR system is about 1% better using the rule-based segmenter, but the rules are not easily extended to continuous speech.

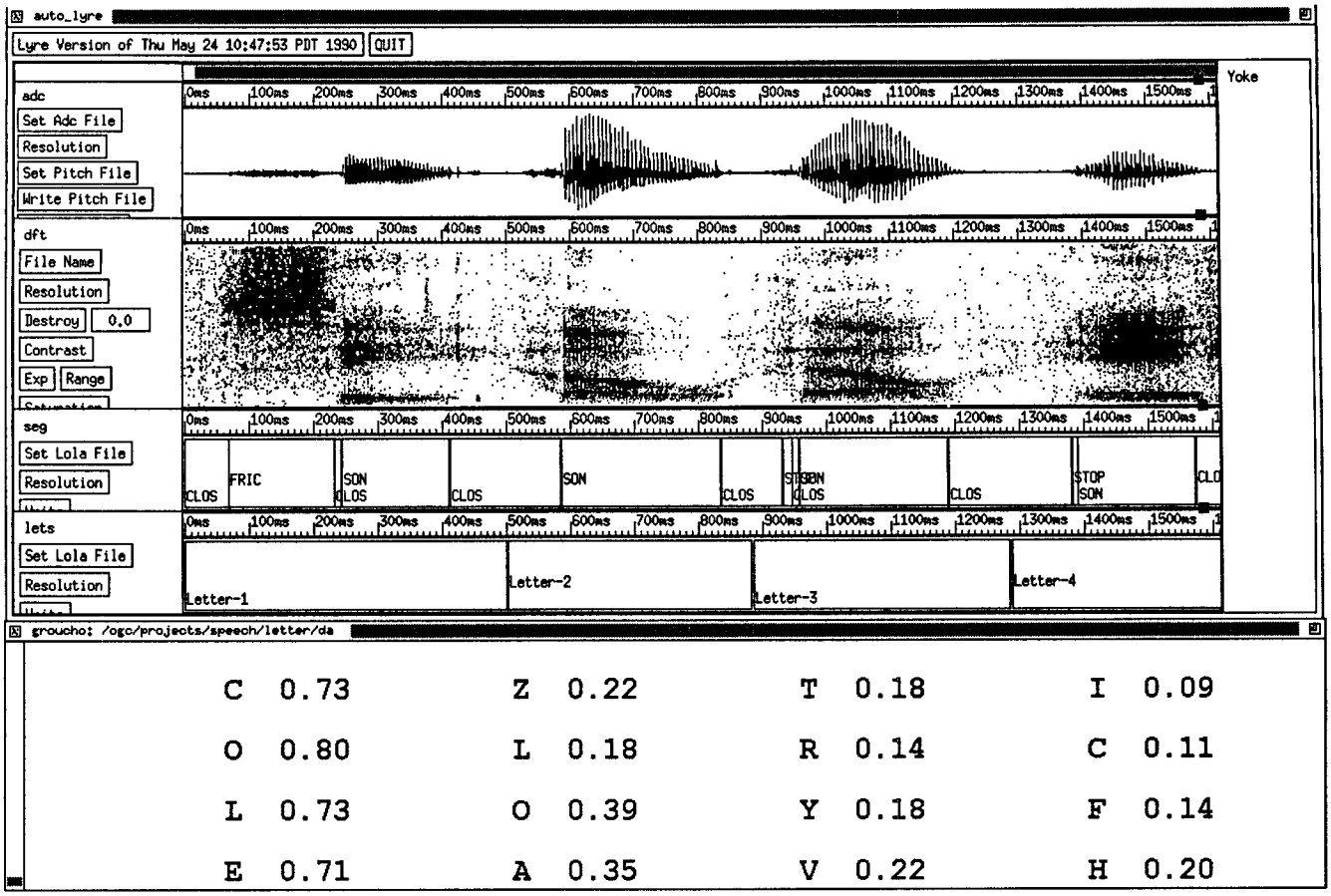

Figure 3: X windows display for the utterance "C O L E." The top four panels show (a) the digitized waveform, (b) the spectrogram, (c) the output of the segmenter, and (d) the location of the letters. The lower panel shows the classification performance of the system. Each row has the top four system outputs for a letter.

sists of a fully connected feed–forward net with 244 input units, 16 hidden units and 4 output units. The segmenter produces an output every 3 msec for each broad category label. The frame-by-frame output of the classifier is converted to a string of broad category labels by taking the largest output value at each time frame after applying a 5-point median smoothing to the outputs across successive frames. Simple duration rules are then applied to the resulting string to prevent short spurious segments, such as a SON less than 80 msec.

The network was trained on multiple letter strings produced by 30 male and 30 female speakers. The features used to train the network consist of the spectrum for the frame to be classified, and the waveform and spectral difference parameters in a 300 msec window centered on the frame. The features were designed to provide detailed information in the immediate vicinity of the frame and less detailed information about the surrounding context.

Letter Segmentation

Letter segmentation is performed by applying rules to the sequence of broad category labels produced by the neural network. The rules are relatively simple because the speakers are required to pause between letters. Except for W, all letters have a single sonorant segment. We assume every sonorant is part of a distinct letter. The boundary between adjacent sonorants is placed in the center of the last closure before the second sonorant. In the English alphabet, all within-letter closures occur after the sonorant (i.e. X and H), so these simple rules capture every case except W, which is usually realized as two sonorants. Our system usually treats W as two letters; we recover from this over-segmentation during the search process.

Letter Classification

A single fully connected feed–forward network with 617 inputs, 52 hidden units and 26 outputs was used to classify letters. This is similar to the first network used in EAR, although spectral coefficients were based on a 128-point DFT. The network was trained on a combination of data from the ISOLET database and 60 additional speakers spelling names and random strings with pauses.

Name Retrieval

After the individual letters are classified, a database of names is searched to find the best match. For this search, the values of the 26 output units are used as the scores

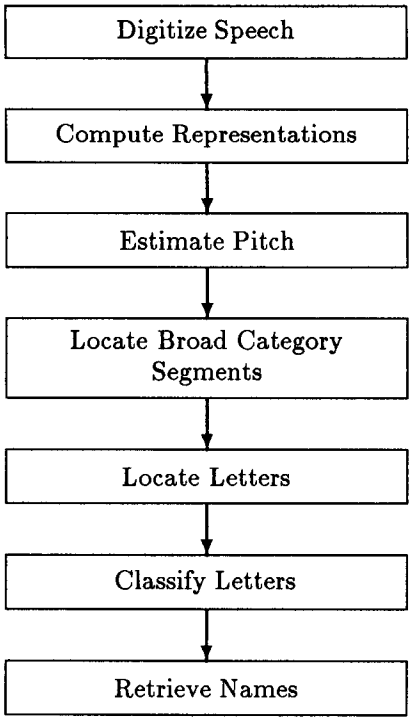

Figure 2: Name Retrieval Modules

for the 26 letters. For each letter classified, 26 scores are returned. The score for a name is equal to the product of the scores returned for the letters in that name in the corresponding positions.

The number of letters found may not match the number of letters in the target name for a number of reasons: there can be segmentation errors; non-letter sounds can be mistaken for letters; the name can be mis-spelled. Because of such errors, letters inserted into or deleted from names are penalized but do not invalidate a match.

We deal with split Ws during name retrieval in the follow way. If a string of letters does not score well against any name, then all pairs of letters in the string for which the second letter is U are collapsed into a W with a score of 0.5 and the search is repeated. This trick has worked surprisingly well in our initial studies because the second part of W is almost always classified as U and because replacing W in a name with something–U does not usually yield another name. Future systems will deal with W in a more elegant manner.

Results

We tested our system on 100 spelled names from 10 new speakers. Name retrieval was evaluated on two databases. The first database consisted of 10,940 names from a local mailing list. Ignoring split Ws (which caused no name-retrieval errors), 697 of 719 letters (97%) were correctly located. Of these, 95.7% were correctly classified. The correct name was returned as the first choice 97 of 100 times. For the three errors, the correct name was the second or third choice twice. (The other name contained three letters spoken without pause. We did not strictly screen the database for pauses because we wanted some borderline cases as well.) Sixty-eight of the one-hundred names were also in a database of 50,000 common last names. When using this database, 65 of 68 names were returned as the first choice. The correct name was the second choice for the 3 errors.

Discussion

English alphabet recognition has been a popular task domain in computer speech recognition for a number of years. Early work, reviewed in [7], applied dynamic programming to frame by frame matching of input and reference patterns to achieve speaker-dependent recognition rates of 60% to 80%. A substantial improvement in recognition accuracy was demonstrated in the FEATURE system, which combined knowledge-based feature measurements and multivariate classifiers to obtain 89% speaker-independent recognition of spoken letters [5]. In recent years, increased recognition accuracy, to a level of 93%, has been obtained using hidden Markov models [8, 9].

It is difficult to compare recognition results across laboratories because of differences in databases, recording conditions, signal bandwidth, signal to noise ratio and experimental procedures. Still, as Table 2 reveals, performance of the EAR system compares favorably to previously reported systems.

We attribute the success of the EAR system to the use of speech knowledge to design features that capture the important acoustic–phonetic information, and the ability of neural network classifiers to use these features to model the variability in the data. Our research has clearly shown that the addition of specific features for difficult discriminations, such as B vs. V, improves recognition accuracy. For example, networks trained with spectral features alone perform about 10% worse than networks trained with the complete set of features.

Explicit segmentation of the speech signal is an important feature of our approach. The location of segment boundaries allows us to measure features of the signal that are most important for recognition. For example, the information needed to discriminate B from D is contained in two main regions: the interval extending 20 msec after the release burst and the 15 msec interval after the vowel onset. By locating the stop burst and the vowel onset, we can measure the important features needed for classification and ignore irrelevant variation in the signal.

We are impressed with the level of performance obtained with the neural network segmenter. We believe the algorithm can be substantially improved with additional features, recurrent networks and more training data. Neural network segmenters have the important advantage of being easily retrained for different databases (e.g., telephone speech, continuous speech), whereas rule-based segmenters require substantial hu-

Study	Conditions	Speakers	Approach	Letters	Results
Brown (1987)	20 kHz Sampling 16.4 dB SNR	100 speakers (multi-speaker)	HMM	E-set	92.0%
Euler et al. (1990)	6.67 kHz Sampling (telephone bandwidth)	100 speakers (multi-speaker)	HMM	26 letters + 10 digits + 3 control words	93.0%
Lang et. al (1990)	Brown's data	100 speakers (multi-speaker)	Neural networks	B,D,E,V	93.0%
Cole, Fanty (1990)	16 kHz Sampling 31 dB SNR	120 training 30 test (speaker-independent)	Knowledge-based features and neural networks	26 letters; E-set; B,D,E,V	96.0% 95.0% 94.2%

Table 2: Recent letter classification results

man engineering.

The application of spoken letter recognition to name retrieval is an obvious and important application. Early work with databases of 18,000 names suggested that spelled names are sufficiently unique so that accurate name retrieval could be obtained without accurate letter recognition [10]. One insight we have gained from our experiments with the 50,000 names is that larger databases do require accurate letter recognition to retrieve names. For example, the 3724 4-letter names in our database generate 20,192 pairs that differ by one letter. Of these, 1372 differ by an acoustically similar letter, such as B-D (152), M-N (128), etc. Correct retrieval of these names requires the system to perform fine phonetic distinctions.

References

[1] Lang, K. J., A. H. Waibel, and G. E. Hinton, "A time-delay neural network architecture for isolated word recognition," *Neural Networks*, **3**, pp. 23–43, (1990).

[2] Barnard, E., R. A. Cole, M. P. Vea and F. Alleva, "Pitch detection with a neural-net classifier," *IEEE Transactions on Acoustics, Speech & Signal Processing*, (Accepted for publication), (1991).

[3] Cole, R. A. and L. Hou, "Segmentation and broad classification of continuous speech," *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing* , New York, (April 1988).

[4] Cole, R. A., Y. Muthusamy and M. A. Fanty, "The ISOLET Spoken Letter Database," Technical Report 90–004, Computer Science Department, Oregon Graduate Institute, (1990).

[5] Cole, R. A., R. M. Stern, M. S. Phillips, S. M. Brill, A. P. Pilant, and P. Specker, "Feature-based speaker-independent recognition of isolated English letters," *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, pp. 731-734, (April 1983).

[6] Barnard, E. and D. Casasent, "Image processing for image understanding with neural nets," in *International Joint Conference on Neural Nets*, (1989).

[7] Cole, R. A., R. M. Stern, and M. J. Lasry, "Performing fine phonetic distinctions: Templates vs. features," in *Invariance and Variability of Speech Processes*, ed. J. Perkell and D. Klatt, Lawrence Erlbaum, New York, (1984).

[8] Brown, P. F., "The acoustic-modeling problem in automatic speech recognition," *Doctoral Dissertation*, Carnegie Mellon University, Dept. of Computer Science (1987).

[9] Euler, S. A., B. H. Juang, C. H. Lee, and F. K. Soong, "Statistical segmentation and word modeling techniques in isolated word recognition," in *Proceedings IEEE International Conference on Acoustics, Speech, and Signal Processing*, (1990).

[10] Aldefeld, B., L. R. Rabiner, A. E. Rosenberg and J. G. Wilpon, "Automated directory listing retrieval system based on isolated word recognition," *Proceedings of the IEEE*, **68**, pp. 1364–1378, (1980).