SPEECH 88 Edinburgh
22/26 August 1988

TRAINING FEATURE DETECTORS FOR USE IN AUTOMATIC SPEECH RECOGNITION

I. S. Howard and M. A. Huckvale

Department of Phonetics and Linguistics,
University College London, Gower Street, London WC1E 6BT.

INTRODUCTION

There are many Automatic Speech Recognition (ASR) systems based on the acoustic matching of an unknown word to a set of reference models. For small vocabulary, speaker-dependent recognition, these systems have acceptable performance for commercial application. However, as the size of the vocabulary in these systems increases, the performance decreases due to the increased similarity of words as seen by the acoustic matching. When such systems are extended to many speakers, performance decreases further due to the increase in acoustic variability of the words arising from the different characteristics of the different speakers, for which the reference models have no structural description.

In this paper we look at a method for incorporating into ASR knowledge about the acoustic-phonetic structure of speech *at the level of the acoustic signal.* We propose a method by which simple acoustic parameters are transformed to a domain where phonetic characteristics of the signal are emphasised. We thus address the two problems with template-based ASR systems described above, since (i) intra-speaker phonetic variability is smaller than acoustic variability, and (ii) the acoustic to phonetic mapping incorporates a simple model of speaker normalization.

The paper consists of four main sections: firstly the justification for the phonetic transformation is argued from a pattern-recognition viewpoint. This is followed by a description of our particular transformation algorithm and its training process, and we then give details of the performance of the transformation in absolute and relative terms. Finally we conclude with some observations about the future of the technique and our approach to ASR.

JUSTIFICATION FOR PHONETIC TRANSFORMATION

All speech recognition systems make assumptions about the acoustic properties of the speech signal that are useful for discriminating words in the vocabulary. They can be said to contain an implicit model of speech production [1], and in particular a model of the range of allowed acoustic variability of input words. The manner in which an ASR system deals with variability is a function of the acoustic parameter set resulting from the front-end processing combined with characteristics of the matching algorithm. Thus a Dynamic Programming (DP) recognizer operating with a Euclidean metric on filterbank energies will recognise correctly only a certain range of acoustic realisations of its target vocabulary. The time-variability it will accept relates to the parameters of the DP time-alignment, while the acoustic variability relates to the behaviour of the Euclidean metric.

With such an analysis, one may criticise a speech recognition algorithm by showing not only what natural variations in pronunciation cause the system to fail, but also what unnatural variations of a word will be recognised correctly.

TRAINING FEATURE DETECTORS FOR USE IN AUTOMATIC SPEECH RECOGNITION

The central argument of this paper is based on the weakness of the case for a Euclidean metric for measuring the acoustic similarity of parts of a speech signal described in terms of low-level acoustic parameters. Although the time-variability model of DP and hidden-Markov Model (HMM) systems is also suspect, we do not address this issue here.

A classifier based on the Euclidian distance metric has limited capabilities compared to a more general scheme, unless a large number of prototype reference patterns are available [2]. As a similarity measure it indicates the distance to a reference point in the feature space. In many ASR implementations, the feature space results from simple analyses of the signal and such analyses often incorporate little knowledge concerning the nature of the speech signal. Many authors have attempted to adapt a set of acoustic parameters to give better recognition performance [recently: 3,4,5] but these have not been motivated by a phonetic model of speech production. We believe our approach to be qualitatively different in that we explicitly aim to generate a set of acoustic-phonetic parameters selected from a *phonological* analysis of a recognition vocabulary. We propose to use the output of a bank of acoustic-phonetic feature detectors, that will give us, for each successive time-window on the signal, an acoustic-phonetic feature vector. Thus instead of acoustic parameter *n* referring to, say, the third cepstral coefficient, it will refer instead to, say, whether or not the speech is nasalised.

This attempt to put speech knowledge into the signal processing front-end of speech recognition systems has two important benefits: firstly, the Euclidean metric can be seen to measure the phonetic similarity of speech signals, and secondly that the performance of the front-end can be assessed against a top-down prediction. Furthermore we can use recognition errors to improve the design of the feature-detector bank, i.e. errors are diagnostic of its lack of discriminative power.

We also note that an early transformation of the acoustic data to a phonetic domain could allow the prediction of templates from *a priori* linguistic knowledge - although we believe that this will require a more sophisticated phonetic model of speech production than is currently available. That is, even in the phonetically transformed data, there is considerable variability that needs to be modelled using parameters of speaker characteristics, allophonic variation, phonetic context, speaking rate, etc.

BUILDING A PHONETIC TRANSFORM SYSTEM

We view the transformation from acoustic parameters to phonetic parameters as a pattern recognition problem, for which we propose the use of a trainable pattern classification algorithm: the Multi-Layer Perceptron (MLP) [6]. This approach is only one of many possible pattern-recognition algorithms, but we have found it to be suitable for this task [7] and better than algorithms that make specific assumptions about the distribution of the input parameters [8].

The required MLP transformation is learnt by repeated presentation of input-output vectors, adapting the internal structure of the perceptron to minimise the squared error between the output required and the output produced. To train the MLP classifier as a feature-detector bank, it is necessary to provide a large body of annotated speech data. In our approach, we label each input pattern vector with the required feature detector outputs (in the experiment described below we use approximately 180 000 pattern vectors). This is to be contrasted to neural network systems that attempt to derive a set of salient features directly from the speech data, for example the transforms of [9]. We think such an approach is misguided since the acoustic features suitable for describing a practicable quantity of speech may be quite different from the optimal set for describing speech in general. This has been shown dramatically in the development of statistical models of speech and language [10]. Thus we specify the required selectivity of the feature detectors and utilise measures of recognition performance to iterate (by hand at present) to a useful set of features.

The database used for training and evaluation consists of 8000 isolated digits, being recordings of 400 digits from each of the 'least consistent' twenty speakers of the RSRE 40-speaker digit database. Each digit has been detected, cut from a recording and encoded, using a 19-channel filterbank to the specification of [11]. The resulting data has 19 energies quantised to 16 levels in 20ms frames. The first 30 repetitions of the digits for each speaker was used for training and the last 10 repetitions for evaluation.

Each digit was phonetically annotated by a process of automatic time alignment with one of a set of manually annotated digits (using a DP algorithm [12]). The annotations were then used to generate a time-aligned feature matrix, having one vector per vocoder frame. The chosen features were:

SIL	Absence of speech signal.
FRIC	Presence of frication.
VOC	Presence of voicing.
NAS	Presence of nasality.
VFRIC	Presence of voicing and frication.
S	Presence of / s /-quality frication.
FTH	Presence of / f / or / θ /-quality frication.
?	Presence of a glottalised vowel onset.
KREL	Presence of a velar plɔsive release.
EE	Presence of / i / or / ɪ /-quality vowel.
EH	Presence of / e /-quality vowel.
UH	Presence of / ʌ /-quality vowel.
ER	Presence of / ə /-quality vowel.
AW	Presence of / ɔ /-quality vowel.
UE	Presence of / u / or / ʊ /-quality vowel.

An example of the feature output for the digit "6" is given in Figure 1: item A is the vocoder energies displayed as a grey-scale, item B the time-aligned annotations, and item C the time-aligned feature matrix.

Input to the MLP network consisted of a 60ms window on the training signal, i.e. the input context was three frames, one on either side of the frame to be labelled. There was one hidden layer of 45 units and 15 feature outputs. Adjacent layers were fully interconnected. Training consisted of one pass over 300 digits of each of the 20 speakers to organise the model for all speakers, followed by 6 additional passes over 300 digits per speaker to produce a set of 20 speaker-dependent classifiers. The network configuration and training parameters were arrived at by trial and error and are probably still not optimum.

Our network differs from that specified by [13] in that we have deliberately chosen a redundant set of features rather than have one output per phoneme. Such features are less abstract and can hopefully be determined with high reliability by means of the feature bank. A consequence of the redundancy is that phonemes can then be made to have ranked similarity in terms of the Euclidian metric, whereas an independent set would give equal distances between all phonemes.

Some approaches to ASR using perceptrons use a large MLP network directly on an input word [14]. This approach requires that the internal representation of speech structure be determined automatically by the learning algorithm. We believe that the use of a abstract model of speech, such as a phonemic analysis, can be used to appropriately constrain the system *a priori*. Since the degrees of freedom are consequently reduced, such a system should be easier to train.

RECOGNITION PERFORMANCE

Item D in Figure 1 shows a typical set of feature detector outputs that may be compared with the training set C. Figure 2 shows the same graphs as Figure 1 but for the digit "9".

The raw performance of each feature detector was determined by plotting a *Receiver Operating Characteristic* (ROC) for each model on the set of 100 test digits for each speaker. The point of equal error (miss rate = false-alarm rate) was used to give the following mean percentage feature detection accuracy:

SIL	92.3 +/- 1.3	FRIC	92.3 +/- 1.9
VOC	94.7 +/- 0.9	NAS	90.3 +/- 3.8
VFRIC	71.5 +/- 4.7		
S	93.7 +/- 1.9	FTH	78.9 +/- 5.9
?	94.6 +/- 3.2	KREL	97.9 +/- 3.0
EE	90.1 +/- 3.6	EH	95.6 +/- 2.1
UH	95.7 +/- 2.1	ER	89.1 +/- 4.0
AW	96.7 +/- 1.3	UE	96.0 +/- 1.5

Since the initial annotations are only approximately aligned with the signal, and since we expect the classifiers to perform worse at phonetic element boundaries, the above performance figures probably underestimate the classification accuracy.

The reference recognition algorithm was a standard implementation of the symmetric dynamic programming algorithm with slope constraints P=1 as described in [15]. The recognition experiments used repetition 31 of the digits for each speaker as templates, and repetitions 32 - 40 as test material. In the speaker-dependent condition, the test digits were matched against templates for the same speaker, in the speaker independent condition the test digits were matched against the template set of each of the other 19 speakers in turn.

The reference algorithm performance on the filterbank data without transformation was as follows:

Speaker-dependent condition % :	97.4 +/- 3.6
Speaker-independent condition % :	88.4 +/- 4.1

The reference algorithm performance on the phonetically transformed data, where the templates were also transformed by the classifier was as follows:

Speaker-dependent condition % :	97.1 +/- 3.6

Speaker-independent condition % : 90.4 +/- 4.1

Neither of these are significantly different to the results using the filterbank parameter set.

The reference algorithm performance on the phonetically transformed test data, where the templates contained ideal feature descriptions derived directly from the annotations was as follows:

Speaker-dependent condition % : 94.4 +/- 5.3
Speaker-independent condition % : 93.0 +/- 2.3

The speaker independent condition has significantly better performance (t-test, correlated samples, p = 0.05) than the filterbank parameter set.

CONCLUSIONS

At this stage we are hopeful that the phonetic transformation can be refined to give even better classification performance, leading to higher digit recognition performance. The reason for our optimism is that we are able to use the classification and recognition errors to guide the process of iterative improvement to the transform. We already have tools for analyzing mismatches in terms of the features used in discrimination.

Although there are weaknesses in the DP algorithm for ASR, both in the DP time alignment and in the metric, and this will clearly affect the results given here, we are committed to evolutionary development of speech recognition systems through the gradual incorporation of speech knowledge into existing architectures.

ACKNOWLEDGEMENTS

This work was supported by MOD university research agreement 2047/137 with SRU and by a SERC fellowship. The digit database was used with kind permission of RSRE.

REFERENCES

[1] M A HUCKVALE, 'ASR beyond HMM', *Proc. Euro. Confr. Sp. Tech.*, Edinburgh, (1987).

[2] J T TOU & R C GONZALES, *Pattern recognition principles*, Addison-Wesley, (1974)

[3] H HERMANSKY, 'Automatic speech recognition and human auditory perception', *Proc. Euro. Confr. Sp. Tech.*, Edinburgh, (1987).

[4] J S D MASON & Y GU, 'Perceptually based features in ASR', *IEE colloquium digest 11*, (1988).

[5] Y TOHKURA, 'A weighted cepstral distance measure for speech recognition', *IEEE Trans. ASSP*, vol. 35, No. 10, (1987).

[6] D E RUMELHART, G E HINTON & R J WILLIAMS, 'Learning Internal Representations by Error Propagation', in Rumelhart & McClelland *Parallel distributed processing*, MIT press, (1987).

[7] I S HOWARD & M A HUCKVALE, 'Acoustic phonetic attribute determination using multi-layer perceptrons', *IEE colloquium digest* 11, (1988).

[8] I S HOWARD & M A HUCKVALE, 'The application of adaptive constraint satisfaction networks to acoustic phonetic attribute determination', *Proc. Euro. Confr. Sp. Tech.*, Edinburgh, (1987).

[9] J L ELMAN & D ZIPSER, *Learning the hidden structure of speech*, ICS report 8701, University of California at San Diego, (1987).

[10] F JELINEK, 'The development of an experimental discrete dictation recognizer',*Proc. IEEE*, vol. 73, No. 11 (1985).

[11] J HOLMES,'The JSRU 19-channel vocoder', *IEE Proc.*, vol 127, part F, No. 1, (1980).

[12] R M CHAMBERLAIN & J S BRIDLE, 'ZIP: a dynamic programming algorithm for time-aligning two indefinitely long utterances', *IEEE ICASSP* 816, (1983).

[13] H BOURLARD & C J WELLEKENS, 'Multilayer perceptrons and automatic speech recognition', IEEE First annual international conference on neural networks, San Diego, California, (1987).

[14] S M PEELING, R K MOORE & M J TOMLINSON, 'The multi-layer perceptron as a tool for speech pattern processing research', Proc. IOA, vol.8 part 7, pp307-314, Windermere, (1986).

[15] H SAKOE & S CHIBA, 'Dynamic programming algorithm optimization for spoken word recognition', IEEE Trans. on ASSP, Vol. ASSP 23, 67-72, (1978).

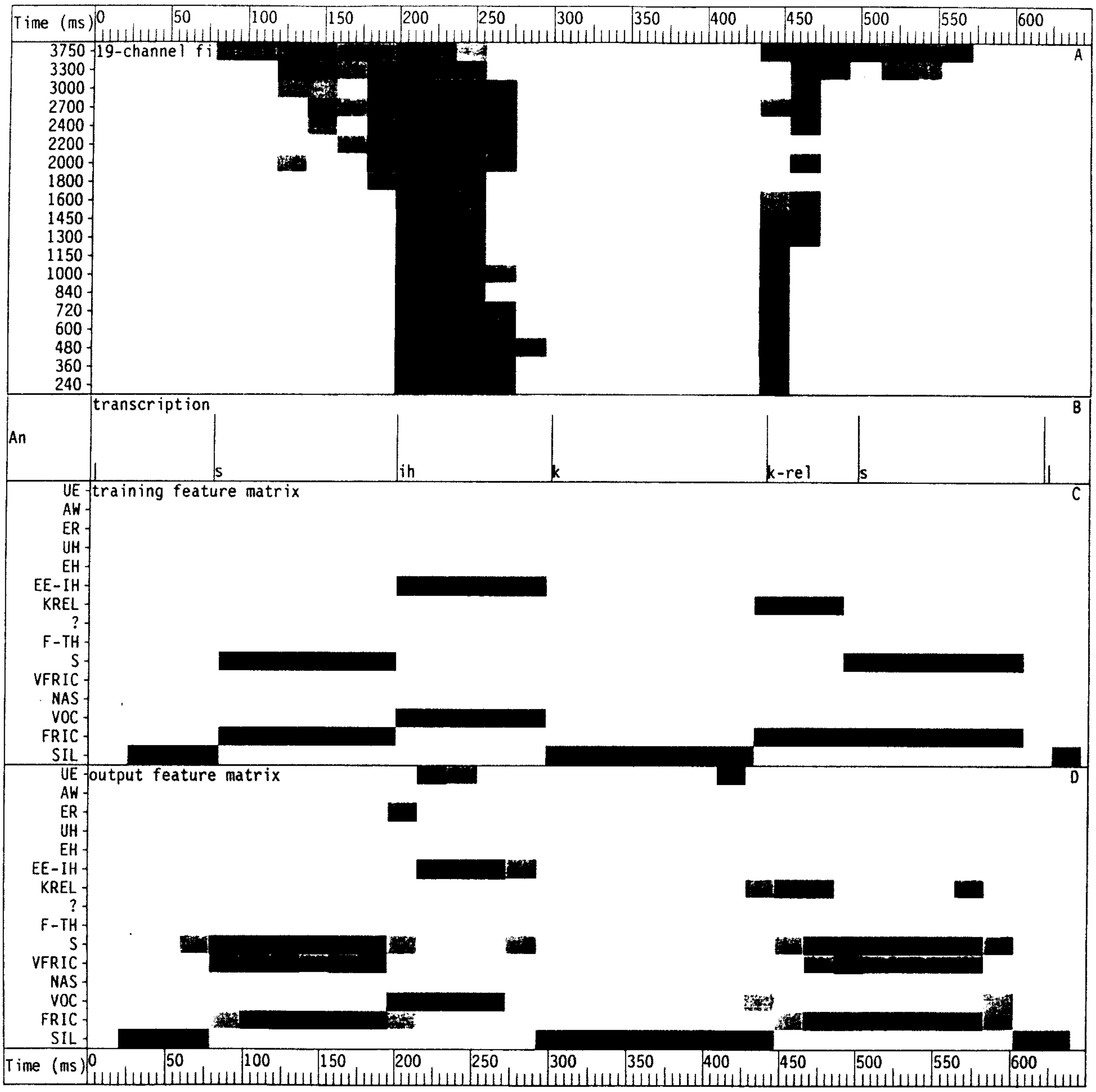

Figure 1. Analyses of the digit "six".

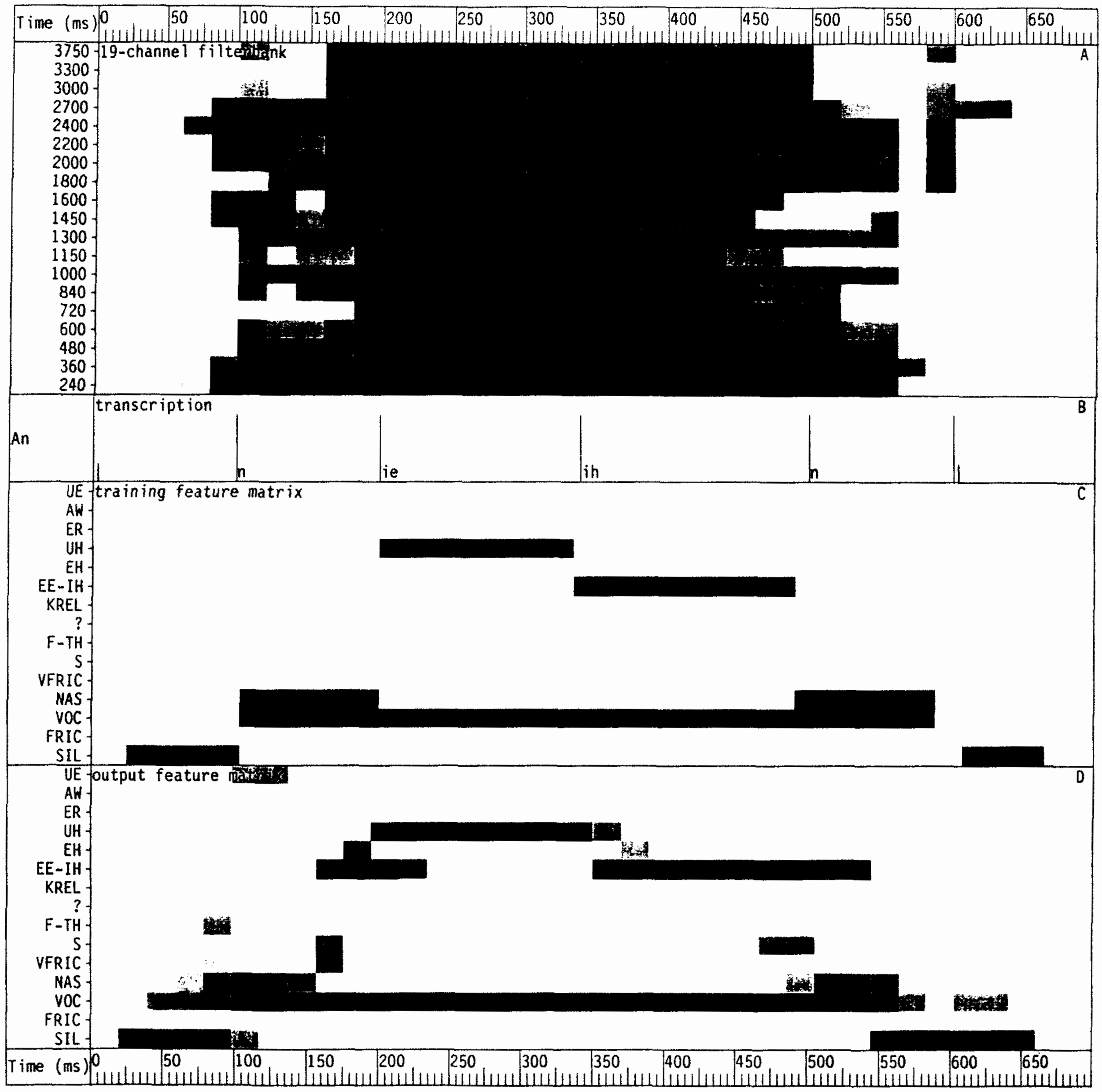

Figure 2. Analyses of the digit "nine".