SPEECH RECOGNITION USING WALSH ANALYSIS AND DYNAMIC PROGRAMMING

P.A.LEE AND J.SEYMOUR

THAMES POLYTECHNIC*

Introduction

The use of Fast Walsh-Hadamard transforms and dynamic programming in the implementation of a small speech recognition system developed around a Z80 microprocessor has been investigated.

Sixteen-point transforms are used which yield a power spectrum of 9 sequency coefficients and for a sampling rate of 10 kHz each transform occupies 1.6 ms. A maximum of twenty transforms produce a timeslot of 32 ms and 32 timeslots occupy 1.024s which defines the maximum length of each utterance. The maximum signal processing frequency is adjustable between 3.4 and 5 kHz.

Comparisons of sinewaves and speech using both Fast Fourier Transforms (FFT) and Fast Walsh-Hadamard Transforms (FWHT) have been produced in this fashion off line by means of a '3-dimensional' graphics package. These show that there are clear similarities in the spectral analyses of speech produced by the two transforms.

For recognition purposes the main features of the speech are extracted by FWHT, which requires only addition and subtraction of samples. Each timeslot is compressed into one byte which is a representation of the spectral power density. A speech pattern always consists of 32 bytes through adjusting the number of transforms in each timeslot. Fifty such patterns can be stored in memory to form a vocabulary.

During recognition a pattern is formed from an input utterance and compared in turn with each master pattern in the vocabulary. An exclusive-OR function is used between bytes and leads to a 32 x 32 similarities matrix. Non-linear time warping is then applied to account for internal time differences and a dynamic programming matrix derived. The number of differences between the two utterances is accumulated and the recognised utterance is the one with the lowest number.

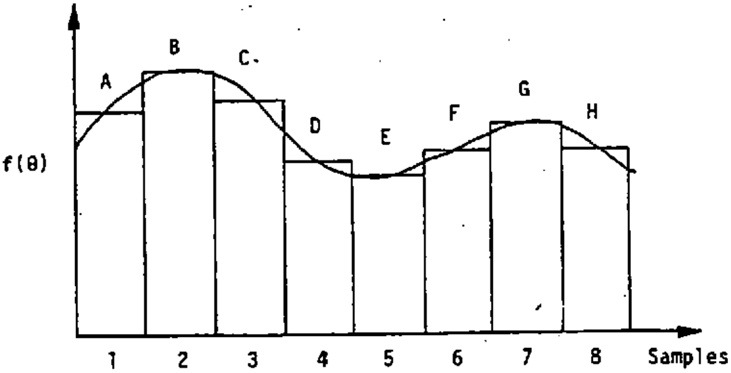
Test results with a panel of speakers indicate that almost 100% recognition can be achieved with the numerals zero to nine and similar results have been obtained with three other vocabularies. Thus Walsh analysis provides a means of extracting the essential parameters of a speech signal and combined with dynamic programming provides an effective method for matching speech patterns, as demonstrated on a small speech recognition system employing a microprocessor in near real-time.

*Now with South Bank Polytechnic

Walsh analysis

For speech analysis by microprocessor using a transform method instead of a filter bank, it is difficult to achieve in real time the complex multiplications of the Fast Fourier Transform FFT. An alternative is the Fast Walsh Hadamard Transform FWHT which uses the same flowchart as the FFT but requires only addition and subtraction of speech samples.

The discrete Walsh coefficients produced, for example, by an 8 point FWHT from data samples lettered A to H are shown in Fig.1. This compares with the 16 coefficients produced by each transform in the speech recognition system. The suffices s and c refer to sal and cal which are odd and even Walsh functions analogous to sine and cosine respectively. The Walsh analogue of frequency is sequency, which is half the total number of zero crossings in the time window and is indicated by the numbers in parentheses.



$$a_c(0) = (+A +B +C +D +E +F +G +H)1/8$$

$$a_s(1) = (+A +B +C +D -E -F -G -H)1/8$$

$$a_c(1) = (+A +B -C -D -E -F +G +H)1/8$$

$$a_s(2) = (+A +B -C -D +E +F -G -H)1/8$$

$$a_c(2) = (+A -B -C +D +E -F -G +H)1/8$$

$$a_s(3) = (+A -B -C +D -E +F +G -H)1/8$$

$$a_c(3) = (+A -B +C -D -E +F -E +H)1/8$$

$$a_s(4) = (+A -B +C -D +E -F +G -H)1/8$$

Fig.1.

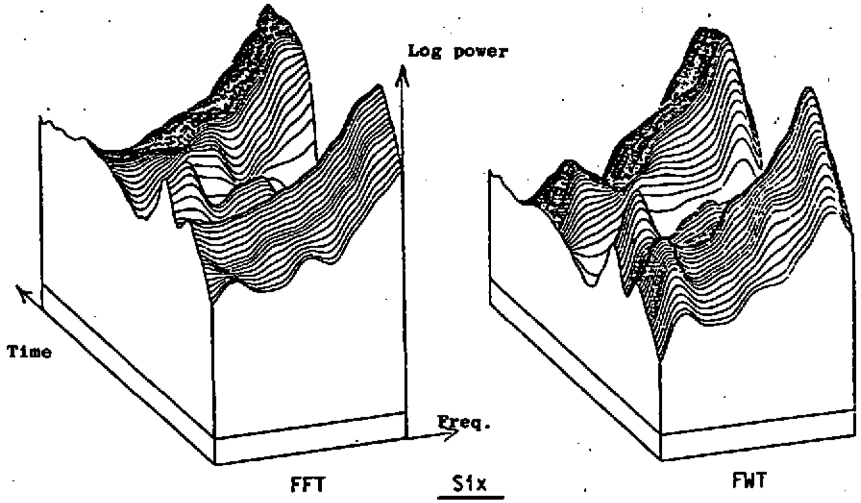


Fig.2.

FFT FEATURES

```
0 0 0 0 1 1 1 1
0 0 1 1 1 1 1 0
0 0 1 1 1 1 1 0
1 0 0 1 1 1 1 0
1 0 1 1 0 0 0 0
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0
0 1 1 0 0 1 1 1
0 1 1 1 0 1 1 0
0 0 1 1 1 1 1 0
0 0 1 1 1 1 1 1
0 0 0 1 1 1 1 1
0 0 0 0 0 0 0 1
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0
```

Six

FWT FEATURES

```
0 0 1 1 1 1 1 0
1 0 1 1 1 0 1 0
1 0 1 1 1 1 1 0
1 0 0 1 1 1 1 0
1 0 1 1 0 0 0 0
0 0 0 1 0 0 0 0
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0
1 1 1 1 1 1 1 1
1 0 0 1 1 1 1 0
1 0 0 1 1 1 1 0
1 0 0 1 1 1 1 0
1 1 0 1 1 1 1 1
0 0 0 0 0 1 0 1
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0
```

Fig.3.

The coefficient values obtained from an FFT are independent of the phase of the input signal relative to the time window. This is not the case with the FWHT and if the input signal changes by a time shift T the coefficients of a particular sequency h will appear at a different sequency k where k = h + T (Beauchamp: 1975). When a sinewave is analysed using the FWHT the coefficients do change in value and sign with phase shift of the input signal, but the changes for a related pair of cal and sal coefficients are such that the sum of $sal^2 + cal^2$ remains constant. Thus the power spectrum coefficients a(k) are independent of phase and are obtained by combining odd and even terms as follows:

$$a(0) = a_c^2\ (0)$$

$$a(k) = a_s^2\ (k) + a_c^2(k)$$

$$a(N/2) = a_s^2\ (N/2)$$

where k = 1, 1, 3, ... (N/2 -1) and N = 16 in this case.

Speech analysis

Speech sampled at 10 kHz and band limited between 200 Hz and 5 kHz was analysed offline on a PRIME computer using both the FFT and the FWHT. The 16 point transforms each occupied 1.6 ms and 40 corresponding power spectra were accumulated. These provided 16 time slots each 64 ms long and 8 frequency slots each 600 Hz wide. The SYMAP and SYMVU packages were then applied to produce a pseudo 3-dimensional effect as shown in Fig.2. for the utterance 'six'. The similarity between the two analysis methods is evident in each case, as is the increased emphasis provided by the FWHT along the utterance. Feature vectors were extracted in a similar manner to that used in the recogniser and are shown in Fig.3. A '1' corresponds to a peak and a '0' to a valley in the spectrum and in general there are more 1's in the FWHT than in the FFT. Further analyses were also carried out with the upper frequency reduced to 4.2 kHz and 3.4 kHz while retaining the 10 kHz sampling rate.

When a sinewave is analysed by the two methods the FFT provides a single component as expected but the FWHT provides several components. These correspond to theoretical predictions and are used to validate the system.

Speech Recognition System

The block diagram of the recogniser is shown in Fig.4. where the upper speech frequency is limited by a switched-capacitor filter, providing a sharp cut-off. This allows adjustment simply by changing the filter clock rate. The lower frequency is fixed at 200 Hz and the active filter includes pre-emphasis at 20 dB per decade. An 8-bit A/D converter operating at 10 kHz passes speech samples into a Z80 microcomputer with a 4MHz clock and 12 K bytes of RAM. All programming is carried out in assembly code for speed of operation, including derivation of the transforms and this contrasts with a previous system in which a hardware Walsh analyser was used (Abu El-Ata and Seymour:1983).

The beginning and end of each utterance are determined by a combination of hardware and software detectors. The timeslot duration has been reduced to 32 ms and time normalisation is achieved by fixing the number of timeslots at

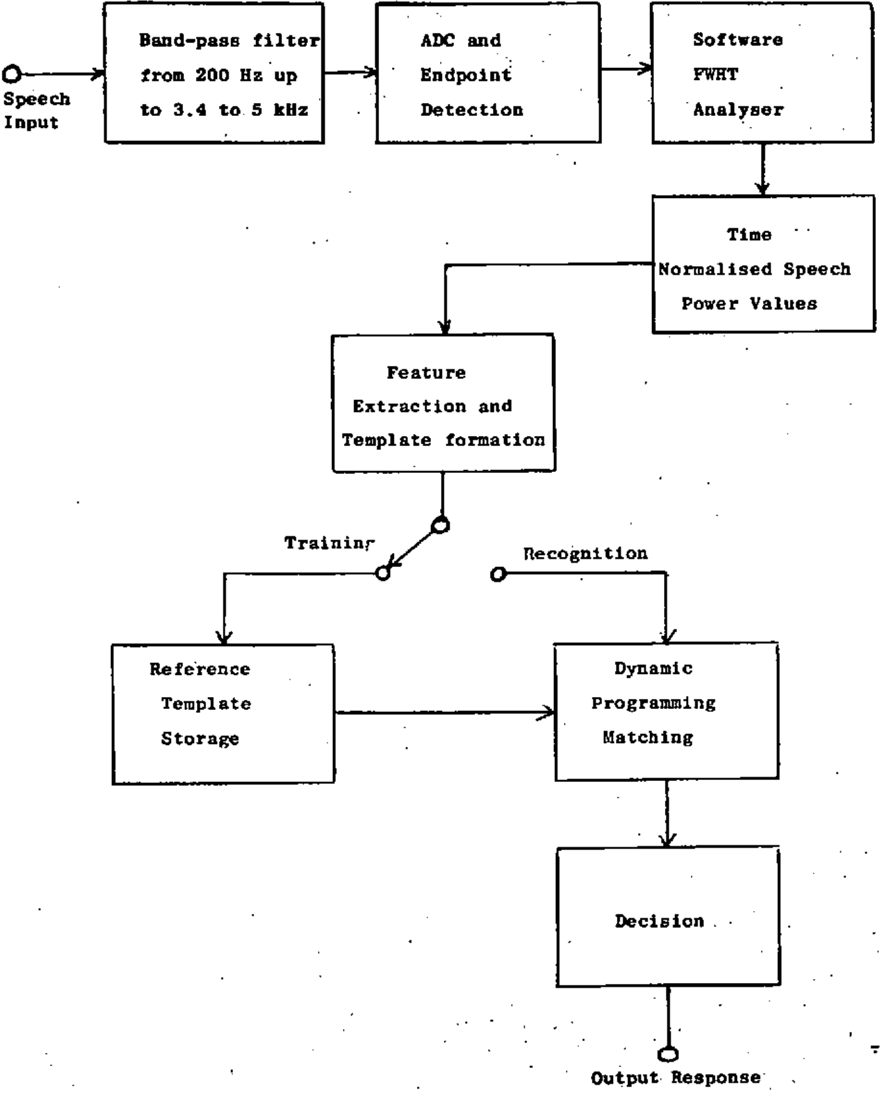

Speech Input
Band-pass filter from 200 Hz up to 3.4 to 5 kHz
ADC and Endpoint Detection
Software FWHT Analyser
Time Normalised Speech Power Values
Feature Extraction and Template formation
Training
Recognition
Reference Template Storage
Dynamic Programming Matching
Decision
Output Response

Fig. 4.

32 and allowing the number of spectra in each one to vary between 1 and 20. This leads to a time restriction of 51.3 ms minimum to 1.024 s maximum duration for any isolated utterance spoken into the system.

The main features in each timeslot are extracted by comparing power coefficients at adjacent sequencies to give a '1' corresponding to a peak and a '0' corresponding to a valley in the spectrum. Each time slot then contains an 8 bit combination or byte corresponding to a feature vector and a speech pattern or template consists of 32 such bytes. The speaker produces one or more patterns for each utterance in the vocabulary, up to 50 of which are stored in memory in the training or learning mode.

Dynamic programming

Time normalisation eliminates the overall time difference between speech templates and also ensures that each timeslot contains a meaningful feature vector. However variation in the rate of speaking leads to non-linear fluctuation along the time axis of a template and so a non-linear time warping function is required to compensate for local compression or expansion of the time scale (Sakoe and Chiba:1978).

Each byte of a pattern A is compared with all the bytes of a pattern B (by means of an exclusive-OR instruction) to create a similarities matrix. This represents the number of differences between the feature vectors at each point s(i,j) in the matrix where:

$$s(i,j) = a_i + b_j$$

$$\text{for all } i = 1 \text{ to } I,\ j = 1 \text{ to } J$$

In the present case I and J are both 32 and notionally A represents a master pattern produced during the training mode while B represents an incoming unknown utterance.

A constant band method of dynamic programming is used (White:1978) in which a dynamic programming matrix D (i,j) is formed from the similarities matrix and is described by the equation.

$$D(i,j) = s(i,j) + \text{Min}\left[D(i,j-1) + P,\ D(i-1,\ j-1),\ D(i-1,j) + P\right]$$
where penalty P = 0, 1, 2, 3 ....

The dynamic programming matrix is constructed by accumulating the raw similarities matrix date of s(i,1) into D(i,1) for i = 1 to I and similarly accumulating s(1,j) into D(1,j) for j = 1 to J. An increasing range of values is formed from D(1,1) in both the i row and the j column respectively.

The matrix is completed for all i and j between i = 2 to I, j = 2 to J. The total dynamic programming distance score DPDS for the template is the value that appears at D(i,j) for i = I, j = J. The DPDS is a measure of the similarity between the two templates representing the utterances.

The optimum value for the penalty P was determined from 1500 tests using the numerals zero to nine. This showed that P = 1 gave the highest recognition score of 97% and this value was used in all subsequent tests.

The closer the value of the DPDS is to zero the closer is the similarity between the two templates. An unknown utterance will be recognised by the master utterance that provides the lowest DPDS from the vocabulary. This is illustrated below for the 10 word vocabulary of the numerals zero to nine when an incoming utterance 'six is compared with each master in turn

Vocabulary	0	1	2	3	4	5	6	7	8	9
DPDS	12	15	18	10	7	9	5	6	9	12
							↑ lowest value			

In the event of two DPDS values being the same the zero crossing counts were compared and the closest taken to represent the recognised utterance.

Results and Conclusions

In evaluating the system four vocabularies each of ten utterances were used as shown in Table 1. Four male and one female speaker each recorded five repetitions of each vocabulary. Each set of utterances was tested against the 50 recordings for a particular vocabulary, producing 1000 individual tests for each speaker.

The results are summarised in Table 2. Tests i to iv were all carried out with single master patterns and test v with from one to four master patterns. Test i showed that the cities provided the best scores and these were used in test ii to investigate bandwidth reduction, which appeared to have a negligible effect on recognition. In test iii all four vocabularies were used with one master pattern from each of 40 utterances and the results are comparable with those in the previous two tests. In test iv a close-talking noise-cancelling microphone was used in a room with background noise at a level between 45 and 70 dBA. Comparable scores were again obtained and it was found that at high noise levels the score was maintained provided that the noise was at a consistent level. Finally, when tests were carried out with an increasing number of master templates on the cities vocabulary the score increased to 100% for five masters. Here the misrecognition caused by a faulty or bad template was eliminated.

In conclusion Walsh analysis and dynamic programming have led to a low cost speech recognition system, without a hardware filter bank and with the analysis and processing of speech achieved in software.

References

1. M.A. ABU EL-ATA and J.SEYMOUR 1983 Acustica 54, 52-56.
A speech recognition system using Walsh analysis with a small computer.

2. K.G. BEAUCHAMP 1975 Academic Press.
Walsh functions and their applications.

3. H.SAKOE and S.CHIBA 1978 IEEE Trans ASSP 26, 43-49.
Dynamic programming algorithms, optimisation for spoken word recognition.

Numerals	Control 1	Control 2	Cities
zero	draw	reverse	Andover
one	go	forward	Birmingham
two	stop	reject	Coventry
three	left	retard	Edinburgh
four	right	increase	Huddersfield
five	up	reduce	Southampton
six	down	higher	Exeter
seven	kill	lower	Manchester
eight	send	accept	Rochester
nine	get	advance	Liverpool

Table 1 Vocabulary Sets

Test No.	Type of input/test	(%) min score	(%) max score	(%) average score
(i)	recorded numerals	91.6	98.0	94.5
	recorded control 1	88.8	95.0	91.1
	recorded control 2	94.4	98.8	95.9
	recorded cities	95.2	97.2	96.3
(ii)	recorded cities bandwidth 5.0kHz	95.2	97.2	96.3
	recorded cities bandwidth 4.2kHz	92.8	98.0	95.2
	recorded cities bandwidth 3.4kHz	93.2	98.0	95.7
(iii)	recorded forty word vocabulary	91.0	95.0	93.0
(iv)	live input numerals	-	-	98.0
	live input cities	-	-	96.0
(v)	recorded cities 1 master	-	-	98.0
	recorded cities 2 masters	-	-	98.4
	recorded cities 3 masters	-	-	99.2
	recorded cities 4 masters	-	-	99.6
	recorded cities 5 masters	-	-	100.0

Table 2 Summary of Results