

SPEECH PROCESSING

This invention relates to speech recognition and in particular the generation of features for use in speech recognition.

5 Automated speech recognition systems are generally designed for a particular use. For example, a service that is to be accessed by the general public requires a generic speech recognition system designed to recognise speech from any user. Automated speech recognisers associated with data specific to a user are used either to recognise a user or to verify a user's claimed identity (so-called
10 speaker recognition).

Automated speech recognition systems receive an input signal from a microphone, either directly or indirectly (e.g. via a telecommunications link). The input signal is then processed by speech processing means which typically divide the input signal into successive time segments or frames by producing an
15 appropriate (spectral) representation of the characteristics of the time-varying input signal. Common techniques of spectral analysis are linear predictive coding (LPC) and Fourier transform. Next the spectral measurements are converted into a set or vector of features that describe the broad acoustic properties of the input signals. The most common features used in speech recognition are mel-frequency cepstral
20 coefficients (MFCCs).

The feature vectors are then compared with a plurality of patterns representing or relating in some way to words (or parts thereof) or phrases to be recognised. The results of the comparison indicate the word/phrase deemed to have been recognised.

25 The pattern matching approach to speech recognition generally involves one of two techniques: template matching or statistical modelling. In the former case, a template is formed representing the spectral properties of a typical speech signal representing a word. Each template is the concatenation of spectral frames over the duration of the speech. A typical sequence of speech frames for a
30 pattern is thus produced via an averaging procedure and an input signal is compared to these templates. One well-known and widely used statistical method of characterising the spectral properties of the frames of a pattern is the hidden Markov model (HMM) approach. The underlying assumption of the HMM (or any

other type of statistical model) is that the speech signal can be characterised as a parametric random process and that the parameters of the stochastic process can be determined in a precise, well-defined manner.

A well known deficiency of current pattern-matching techniques, especially HMMs, is the lack of an effective mechanism for the utilisation of the correlation of the feature extraction. A left-right HMM provides a temporal structure for modelling the time evolution of speech spectral characteristics from one state into the next, but within each state the observation vectors are assumed to be independent and identically distributed (IID). The IID assumption states that there is no correlation between successive speech vectors. This implies that within each state the speech vectors are associated with identical probability density functions (PDFs) which have the same mean and covariance. This further implies that the spectral-time trajectory within each state is a randomly fluctuating curve with a stationary mean. However in reality the spectral-time trajectory clearly has a definite direction as it moves from one speech event to the next.

This violation by the spectral vectors of the IID assumption contributes to a limitation in the performance of HMMs. Including some temporal information into the speech feature can lessen the effect of this assumption that speech is a stationary independent process, and can be used to improve recognition performance.

A conventional method which allows the inclusion of temporal information into the feature vector is to augment the feature vector with first and second order time derivatives of the cepstrum. A mathematically more implicit representation of speech dynamics is the cepstral-time matrix which uses a cosine transform to encode the temporal information as described in B P Milner and S V Vaseghi, "An analysis of cepstral-time feature matrices for noise and channel robust speech recognition", Proc. Eurospeech, pp 519-522, 1995.

A cepstral-time matrix, $c_t(m,n)$, is obtained either by applying a 2-D Discrete Cosine Transform (DCT) to a spectral-time matrix or by applying a 1-D DCT to a stacking of mel-frequency cepstral coefficients (MFCC) speech vectors. M N-dimensional log filter bank vectors are stacked together to form a spectral-time matrix, $X_t(f,k)$, where t indicates the time frame, f the filter bank channel and k the time vector in the matrix. The spectral-time matrix is then transformed into a

This Sheet Has Been Replaced



cepstral-time matrix using a two-dimensional DCT. Since a two-dimensional DCT can be divided into two one-dimensional DCTs, an alternative implementation of the cepstral-time matrix is to apply a 1-D DCT along the time axis of a matrix consisting of M conventional MFCC vectors.

5 According to a first aspect of the invention there is provided a method of generating features for use in speech recognition, said method comprising calculating the log frame energy value of each of a pre-determined number n of frames of an input speech signal; and applying a transform to the n log frame energy values to form a temporal vector representing the input speech signal.

10 Speech transitional dynamics are produced implicitly within the temporal vector, compared to the explicit representation achieved with a cepstral vector with derivatives augmented on. Thus, models trained on such matrices have the advantage that inverse transforms can be applied which allow transforms back into the linear filter bank domain for techniques such as parallel model combination
15 (PMC), for improved noise robustness.

The transform may be a discrete cosine transform. Preferably the temporal matrix is truncated so as to include fewer than n elements. This has been found to produce good performance results whilst reducing the amount of computation involved. The steady state ($m=0$) column of the matrix may be
20 omitted, so removing any distortion of the speech signal by a linear convolutional channel distortion making the matrix a channel robust feature.

The invention also relates to a method of speech recognition including the feature generating method of the invention.

In another aspect of the invention there is provided feature generating
25 apparatus for use in speech recognition comprising:

a processor for calculating the logarithm of the energy of each of a pre-determined number n of frames of an input speech signal; and

a processor for applying a transform to the n log energy values so calculated to form a temporal vector representing the input speech.

30 The feature generating means of the invention is suitable for use with speech recognition apparatus and also to generate recognition data for use with such apparatus.

This Sheet Has Been Replaced



The invention will now be described by way of example only by reference to the accompanying drawings in which:

Figure 1 shows schematically the employment of a speech recogniser in a telecommunications environment;

5 Figure 2 is a schematic representation of a speech recogniser;

Figure 3 shows schematically the components of one embodiment of a feature extractor according to the invention;

Figure 4 shows the steps for determining a Karhunen-Loeve transform;

10 Figure 5 shows schematically the components of a conventional speech classifier forming part of the speech recogniser of Figure 2;

Figure 6 is a flow diagram showing schematically the operation of the classifier of Figure 5;

Figure 7 is a block diagram showing schematically the components of a conventional sequencer forming part of the speech recogniser of Figure 2;

15 Figure 8 shows schematically the content of a field within a store forming part of the sequencer of Figure 7; and

Figure 9 is a flow diagram showing schematically the operation of the sequencer of Figure 7.

Referring to Figure 1, a telecommunications system including speech
20 recognition generally comprises a microphone 1 (typically forming part of a telephone handset), a telecommunications network 2 (typically a public switched telecommunications network (PSTN)), a speech recogniser 3, connected to receive a voice signal from the network 2, and a utilising apparatus 4 connected to the speech recogniser 3 and arranged to receive therefrom a voice recognition signal,
25 indicating recognition or otherwise of a particular word or phrase, and to take action in response thereto. For example, the utilising apparatus 4 may be a remotely operated terminal for effecting banking transactions, an information service etc.

In many cases, the utilising apparatus 4 will generate an audible response
30 to the user, transmitted via the network 2 to a loudspeaker 5 typically forming part of the user's handset.

In operation, a user speaks into the microphone 1 and a signal is transmitted from the microphone 1 into the network 2 to the speech recogniser 3.

The speech recogniser analyses the speech signal and a signal indicating recognition or otherwise of a particular word or phrase is generated and transmitted to the utilising apparatus 4, which then takes appropriate action in the event of recognition of the speech.

5 Generally the speech recogniser 3 is ignorant of the route taken by the signal from the microphone 1 to and through network 2. Any one of a large variety of types or qualities of handset may be used. Likewise, within the network 2, any one of a large variety of transmission paths may be taken, including radio links, analogue and digital paths and so on. Accordingly the speech signal Y
10 reaching the speech recogniser 3 corresponds to the speech signal S received at the microphone 1, convolved with the transform characteristics of the microphone 1, the link to the network 2, the channel through the network 2, and the link to the speech recogniser 3, which may be lumped and designated by a single transfer characteristic H.

15 Typically, the speech recogniser 3 needs to acquire data concerning the speech against which to verify the speech signal, and this data acquisition is performed by the speech recogniser in the training mode of operation in which the speech recogniser 3 receives a speech signal from the microphone 1 to form the recognition data for that word or phrase. However, other methods of acquiring the
20 speech recognition data are also possible.

Referring to Figure 2, a speech recogniser comprises an input 31 for receiving speech in digital form (either from a digital network or from an analog to digital converter); a frame generator 32 for partitioning the succession of digital samples into a succession of frames of contiguous samples; a feature extractor 33
25 for generating a corresponding feature vector from the frames of samples; a classifier 34 for receiving the succession of feature vectors and generating recognition results; a sequencer 35 for determining the predetermined utterance to which the input signal indicates the greatest similarity; and an output port 35 at which a recognition signal is supplied indicating the speech utterance which has
30 been recognised.

As mentioned earlier, a speech recogniser generally obtains recognition data during a training phase. During training, speech signals are input to the speech recogniser 3 and a feature is extracted by the feature extractor 33

according to the invention. This feature is stored by the speech recogniser 3 for subsequent recognition. The feature may be stored in any convenient form, for example modelled by Hidden Markov Models (HMMs), a technique well known in speech processing, as will be described below. During recognition, the feature
 5 extractor extracts a similar feature from an unknown input signal and compares the feature of the unknown signal with the feature(s) stored for each word/phrase to be recognised.

For simplicity, the operation of the speech recogniser in the recognition phase will be described below. In the training phase, the extracted feature is used
 10 to train a suitable classifier 34, as is well known in the art.

Frame Generator 32

The frame generator 32 is arranged to receive speech samples at a rate of, for example, 8,000 samples per second, and to form frames comprising 256
 15 contiguous samples, at a frame rate of 1 frame every 16ms. Preferably, each frame is windowed (i.e. the samples towards the edge of the frame are multiplied by predetermined weighting constants) using, for example, a Hamming window to reduce spurious artefacts generated by the frame edges. In a preferred embodiment, the frames are overlapping (for example by 50%) so as to ameliorate
 20 the effects of the windowing.

Feature Extractor 33

The feature extractor 33 receives frames from the frame generator 32 and generates, from each frame, a feature or vector of features. Figure 3 shows an
 25 embodiment of a feature extractor according to the invention. Means may additionally be provided to generate other features, for example LPC cepstral coefficients or MFCCs.

Each frame j of an incoming speech signal is input to a processor 331 which calculates the average energy of the frame of data, i.e. the energy calculator
 30 processor 331 calculates:

$$E_{av,j} = \frac{1}{256} \sum_{i=1}^{256} x_i^2$$

where x_i is the value of the energy of sample i in frame j .

A logarithmic processor 332 then forms the log of this average value for the frame j . The log energy values are input into a buffer 333 which has a length sufficient to store the log energy values for n frames e.g. $n=7$. Once seven frames' worth of data has been calculated the stacked data is output to a
 5 transform processor 334.

In the formation of the frame energy vector or temporal matrix the spectral-time vector of the stacked log energy values input to the transform processor 334 is multiplied by a transform matrix, i.e.

$$MH = T$$

10 where M is the vector of stacked log energy values, H is the transform which can encode the temporal information, and T is the frame energy vector.

The columns of the transform H are the basis functions for encoding the temporal information. Using this method of encoding temporal information, a wide range of transforms can be used as the temporal transform matrix, H .

15 The transform H encodes the temporal information, i.e. the transform H causes the covariance matrix of the log energy value stack to be diagonalised. That is to say, the elements of the off-diagonal (i.e. the non-leading diagonal) of the covariance matrix of the log energy values transformed by H tend to zero. The off-diagonal of a covariance matrix indicates the degree of correlation between
 20 respective samples. The optimal transform for achieving this is the Karhunen-Loeve (KL) transform as described in the book by N S Jayant and P Noll, "Digital coding of waveforms", Prentice-Hall, 1984.

To find the optimal KL transform for encoding the temporal information conveyed by the feature vectors, statistics regarding the successive correlation of
 25 the vectors is needed. Using this correlation information, the KL transform can then be calculated. Figure 4 shows the procedure involved in determining the KL transform from speech data.

To accurately determine the KL transform the entire set of training data is first parameterised into log energy values. Vectors x_t , containing n successive log
 30 energy values in time, are generated:

$$x_t = [c_t, c_{t-1}, \dots, c_{t+n-1}]$$

From the entire set of these vectors across the training set, a covariance matrix, Σ_{xx} , is calculated $\Sigma_{xx} = E\{xx^T\} - \mu_x \mu_x^T$, where μ_x is the mean vector of the log energy values.

As can be seen, this is closely related to the correlation matrix, $E\{xx^T\}$, and as such contains information regarding the temporal dynamics of the speech. The KL transform is determined from the eigenvectors of the covariance matrix, and can be calculated, for example using singular value decomposition, where,

$$H^T \Sigma_{xx} H = \text{dia}(\lambda_0, \lambda_1, \dots, \lambda_M) = \Lambda$$

The resulting matrix, H , is made up from the eigenvectors of the covariance matrix. These are ranked according to the size of their respective eigenvalues, λ_i . This matrix is the KL-derived temporal transform matrix.

Other polynomials can be used to generate the temporal transform matrix, such as Legendre, Laguerre etc. The KL transform is complicated by the need to calculate the transform itself for each set of training data. Alternatively the Discrete Cosine Transform (DCT) may also be used. In this case, the transform processor 334 calculates the DCT of the stacked data relating to the log energy values for n frames.

The one-dimensional DCT is defined as:

$$F(u) = \sqrt{\frac{2}{n}} C(u) \sum_{i=0}^{n-1} f(i) \cos \left[\frac{(2i+1)u\pi}{2n} \right]$$

where

$f(i)$ = log energy value for frame i

$C(u) = 1/\sqrt{2}$ for $u=0$

= 1 otherwise

u is an integer from 0 to $n-1$

The transform processor 334 outputs n DCT coefficients generated from n frames of data. These coefficients form a frame-energy vector relating to the energy level of the input signal.

A frame energy vector is formed for each successive n frames of the input signal e.g. for frames 0 to 6, 1 to 7, 2 to 8 and so on when $n=7$. The frame energy vector forms part of a feature vector for a frame of speech. This feature may be used to augment other features e.g. MFCCs or differential MFCC.

Classifier 34

Referring to Figure 5, the classifier 34 is of a conventional design and, in this embodiment, comprises a HMM classifying processor 341, an HMM state
5 memory 342, and a mode memory 343.

The state memory 342 comprises a state field 3421, 3422, ..., for each of the plurality of speech parts to be recognised. For example, a state field may be provided in the state memory 342 for each phoneme of a word to be recognised. There may also be provided a state field for noise/silence.

10 Each state field in the state memory 342 includes a pointer field 3421b, 3422b, storing a pointer address to a mode field set 361, 362, in mode memory 343. Each mode field set comprises a plurality of mode fields 3611, 3612... each comprising data defining a multidimensional Gaussian distribution of feature coefficient values which characterise the state in question. For example, if
15 there are d coefficients in each feature (for instance the first 8 MFCC coefficients and the seven coefficients of the energy-matrix of the invention), the data stored in each mode field 3611, 3612... characterising each mode is: a constant C , a set of d feature mean values μ_i and a set of d feature deviations, σ_i ; in other words, a total of $2d + 1$ numbers.

20 The number N_i of mode fields 3611, 3612, in each mode field set 361, 362, is variable. The mode fields are generated during the training phase and represent the feature(s) derived by the feature extractor.

During recognition, the classification processor 34 is arranged to read each state field within the memory 342 in turn, and calculate for each, using the current
25 input feature coefficient set output by the feature extractor 33 of the invention, the probability that the input feature set or vector corresponds to the corresponding state. To do so, as shown in Figure 6, the processor 341 is arranged to read the pointer in the state field; to access the mode field set in the mode memory 343 to which it points; and, for each mode field j within the mode
30 field set, to calculate a modal probability P_{ij} .

Next, the processor 341 calculates the state probability by summing the modal probabilities P_{ij} . Accordingly, the output of the classification processor 341

is a plurality of state probabilities P , one for each state in the state memory 342, indicating the likelihood that the input feature vector corresponds to each state.

It will be understood that Figure 6 is merely illustrative of the operation of the classifier processor 341. In practice, the mode probabilities may each be
 5 calculated once, and temporarily stored, to be used in the calculation of all the state probabilities relating to the phoneme to which the modes correspond.

The classifying processor 341 may be a suitably programmed digital signal processing (DSP) device and may in particular be the same digital signal processing device as the feature extractor 33.

10 Sequencer 35

Referring to Figure 7, the sequencer 35 is conventional in design and, in this embodiment, comprises a state probability memory 353 which stores, for each frame processed, the state probabilities output by the classifier processor 341; a state sequence memory 352; a parsing processor 351; and a sequencer output
 15 buffer 354.

The state sequence memory 352 comprises a plurality of state sequence fields 3521, 3522,, each corresponding to a word or phrase sequence to be recognised consisting, in this example, of a string of phonemes. Each state sequence in the state sequence memory 352 comprises, as illustrated in Figure 8,
 20 a number of states $P_1, P_2, \dots, P_N$ and, for each state, two probabilities; a repeat probability (P_{i1}) and a transition probability to the following state (P_{i2}). The observed sequence of states associated with a series of frames may therefore comprise several repetitions of each state P_i in each state sequence model 3521 etc; for example:

Frame Number	1	2	3	4	5	6	7	8	9	...	Z	Z+1
State	P1	P1	P1	P2	P2	P2	P2	P2	P2	...	Pn	Pn

25 As shown in Figure 9 the sequencing processor 351 is arranged to read, at each frame, the state probabilities output by the classifier processor 341, and the previous stored state probabilities in the state probability memory 353, and to calculate the most likely path of states to date over time, and to compare this with each of the state sequences stored in the state sequence memory 352.

30 The calculation employs the well known Hidden Markov Model method described generally in "Hidden Markov Models for Automatic Speech Recognition:

theory and applications" S.J. Cox, British Telecom Technology Journal, April 1988 p105. Conveniently, the HMM processing performed by the sequencing processor 351 uses the well known Viterbi algorithm. The sequencing processor 351 may, for example, be a microprocessor such as the IntelTM i-486TM microprocessor or
5 the MotorolaTM 68000 microprocessor, or may alternatively be a DSP device (for example, the same DSP device as is employed for any of the preceding processors).

Accordingly for each state sequence (corresponding to a word, phrase or other speech sequence to be recognised) a probability score is output by the
10 sequencing processor 351 at each frame of input speech. For example the state sequences may comprise the names in a telephone directory. When the end of the utterance is detected, a label signal indicating the most probable state sequence is output from the sequencing processor 351 to the output port 38, to indicate that the corresponding name, word or phrase has been recognised.

**CLAIMS**

1. A method of generating features for use with speech responsive apparatus, said method comprising:
 - 5 calculating the logarithmic frame energy value of each of a pre-determined number n of frames of an input speech signal; and applying a transform matrix to the n logarithmic frame energy values to form a temporal matrix representing the input speech signal.
- 10 2. A method according to Claim 1 wherein successive features represents overlapping groups of n frames of the input signal.
3. A method according to claim 1 or 2 wherein the transform matrix is a discrete cosine transform.
- 15 4. A method according to claim 1, 2 or 3, wherein the temporal matrix is truncated so as to include fewer than n elements.
5. A method of speech recognition comprising:
 - 20 receiving an input signal representing speech, said input signal being divided into frames;
generating a feature by calculating the logarithmic frame energy value of each of a pre-determined number n of frames of an input speech signal; and applying a matrix transform to the n logarithmic frame energy values to form a
 - 25 temporal matrix representing the input speech signal;
comparing the generated feature with recognition data representing allowed utterances, said recognition data relating to the feature; and
indicating recognition or otherwise on the basis of the comparison step.
- 30 6. A method of speech recognition according to Claim 5 wherein the transform processor is arranged to apply a discrete cosine transform to the n logarithmic energy values.



7. Feature generating apparatus for use with speech responsive apparatus, said feature generating apparatus comprising:
- a processor for calculating the logarithm of the energy of each of a pre-determined number n of frames of an input speech signal; and
- 5 a processor for applying a transform to the n logarithmic energy values so calculated to form a temporal matrix representing the input speech.
8. Feature generating apparatus according to Claim 7 wherein the transform processor is arranged to apply a discrete cosine transform to the n logarithmic
- 10 energy values.
9. Speech recognition apparatus including feature generating apparatus according to claim 7 or 8.
- 15 10. Speech recognition apparatus including an input for receiving recognition data relating to features generated according to claim 1.



This Sheet Has Been Replaced

