EUROSPEECH '97
5th European Conference on Speech
Communication and Technology
Rhodes, Greece, September 22-25, 1997

CEPSTRAL-TIME MATRICES AND LDA FOR IMPROVED CONNECTED DIGIT AND SUB-WORD RECOGNITION ACCURACY

Ben Milner

ben@saltfarm.bt.co.uk

Speech Technology Unit, BT Laboratories,

Martlesham Heath, Suffolk, UK.

ABSTRACT

Previous work has shown that good accuracy improvements can be made for isolated word recognition using cepstral-time matrices as the speech feature instead of the more conventional MFCC-based speech feature augmented with higher order cepstrum. This work extends the performance improvements to UK English connected digit strings and to a sub-word based town names task.

Experimental results are presented for a range different sized cepstral-time matrix widths - ranging from a stack width of 3 up to 13 MFCC frames. In addition a variety of columns are selected from the cepstral-time matrix for use as the final speech feature. Tests show that the optimal implementation of the cepstral-time matrix varies according to the specific recognition task.

Finally the technique of linear discriminative analysis (LDA) is applied to cepstral-time matrices and is shown to successfully improve recognition performance, as well as reducing the size of the final speech feature. Three different implementations of LDA are described and are demonstrated on isolated digit and sub-word tasks.

1. INTRODUCTION

After feature extraction, successive feature vectors are correlated. However a well known deficiency of HMMs is the lack of an efficient mechanism for the utilisation of this correlation. The 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). Including some temporal information into the feature vector can lessen the effect of the IID assumption of HMMs [1]. The conventional way of including temporal information into the speech feature is to augment the cepstrum with the differential cepstrum.

Previous work, [2], has shown that the cepstral-time matrix offers superior performance over conventional speech features such as MFCCs augmented with velocity and acceleration cepstrum [3] for isolated word tasks. Additionally the cepstral-time matrix is inherently robust to channel distortion and offers increased noise robustness over MFCCs.

In this work cepstral-time matrices are applied to both connected digits and a sub-word based town names test. An analysis of the cepstral-time matrix is made in section 2.1. In particular the usefulness of the various regions of the matrix are examined for the connected digit task and a sub-word task. The technique of linear discriminant analysis (LDA) is combined with cepstral-time matrices in section 2.2., where three different implementations are demonstrated. Results are given in section 3 which show experimentally the usefulness of various columns of the cepstral-time matrix for isolated and connected digits as well as a sub-word task. Results for LDA are presented for both the isolated digits and sub-word town names task.

2. CEPSTRAL-TIME MATRICES

A cepstral-time matrix, $C_t(m,n)$, is obtained by applying a 2-D Discrete Cosine Transform (DCT) to a log spectral-time matrix. Since a 2-D DCT can be decomposed into two 1-D DCTs, the cepstral time matrix can also be obtained by applying a 1-D DCT to a stacking of M successive MFCC speech vectors, $c_t(n)$, [2],

$$C_t(m, n) = \sum_{k=0}^{M-1} c_{t+k}(n) \cos\frac{(2k+1)m\pi}{2M} \qquad (1)$$

Figure 1 illustrates the generation of the cepstral-time matrix from a stacking of MFCC speech vectors.

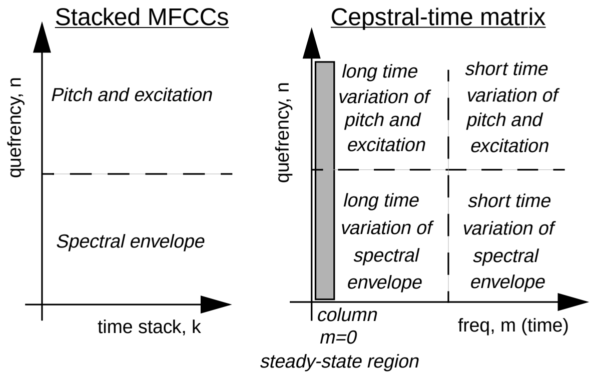
Stacked MFCCs
quefrency, n
Pitch and excitation
Spectral envelope
time stack, k
Cepstral-time matrix
quefrency, n
long time variation of pitch and excitation
short time variation of pitch and excitation
long time variation of spectral envelope
short time variation of spectral envelope
column m=0
steady-state region
freq, m (time)

Figure 1: Regions of the cepstral-time matrix.

In the cepstral-time matrix the lower index coefficients along the quefrency axis, n, represents the spectral envelope, whereas the higher coefficients represent the pitch and excitation, as is the case for the cepstrum. Along the pseudo-time axis, m, the lower coefficients represent the longer time variation of the cepstral coefficients, and the higher coefficients the short time variation. The column $m=0$ contains the average or steady-state level of the stacked MFCCs. Any stationary or slow moving channel distortion present in the speech signal will be compressed into this zeroth column. By removing this column from the speech feature, these channel effects will be removed from the speech feature.

Typically stack widths of between 5 and 11 frames are used, with the first few columns of the resulting cepstral-time matrix forming the speech feature. The choice of stack width and the subsequent columns for selection as the speech feature are discussed in section 3.1.

2.1. Columns of the cepstral-time matrix

Each element of the cepstral-time matrix is generated using a DCT as shown by equation (1). This is equivalent to measuring the correlation between the time-sequence of MFCCs with a cosine basis function comprising a given number of cycles across the time stack.

Each basis function has an associated frequency - the frequency of the cosine wave. The zeroth column represents a frequency of 0Hz as the basis function is a d.c. level. The effective frequency of the remaining columns is dependent upon both the overall stack width of the cepstral-time matrix and the frame rate of the original cepstral analysis. Table 1 shows the analysis frequency of each of the columns for a selection of cepstral-time matrices with varying stack widths. The cepstral analysis was taken every 16ms which corresponds to a 62.5Hz frame rate.

	col 0	*col 1*	*col 2*	*col 3*	*col 4*	*col 5*	*col 6*	*col 7*	*col 8*
width 3	0	15.6	31.2						
width 5	0	7.8	15.6	23.4	31.2				
width 7	0	5.2	10.4	15.6	20.8	26.0	31.2		
width 9	0	3.9	7.8	11.7	15.6	19.5	23.4	27.3	31.2

Table 1: Frequency of each column of the cepstral-time matrix for varying stack widths. (Figures in Hz)

All cepstral-time matrices, independent of their stack width, have 0Hz as their analysis frequency in the zeroth column and 31.25 Hz as the last column.The frequency resolution within the matrix is finer with a wider stack width. For example, column 1 from stack width 3, column 2 from width 5, column 3 from width 7 and column 4 from width 9 all have a basis function frequency of 15.6 Hz. This is illustrated in Figure 2.

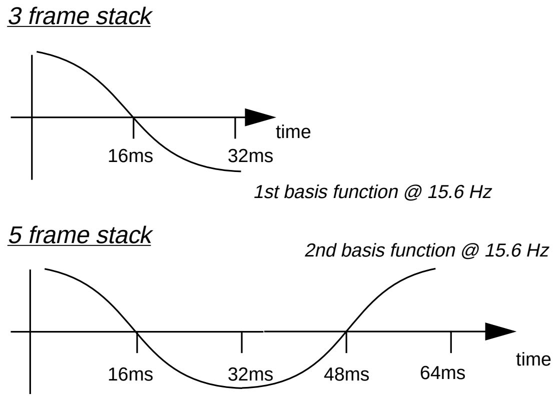

Figure 2: Illustration of frequency of basis function for different stack widths.

2.2. Cepstral-time matrices and LDA

The technique of linear discriminant analysis (LDA) has been successfully used to both improve speech recognition performance and reduce the overall size of the feature vector and hence reduce computation costs, [4]. This can be applied to cepstral-time matrices. The choice of speech class is a fundamental issue in LDA with no ideal choice being obvious. In this work three different choices of speech class are made, based on the states contained within a vocabulary of HMMs.

Considering the application of LDA to a monophone-based recognition task using 3-state HMMs. Each monophone is initially trained on the base speech feature using 3-state HMMs. The most simple allocation of speech class is to treat each state of each HMM in the recogniser vocabulary as a class - Figure 3a. From each class the between-class and within-class covariances can be calculated from the state means and covariances. Using these covariances the LDA transform matrix is computed [5]. This is used to transform the speech vectors into the new feature space which is then typically followed by a truncation. The second method, shown in Figure 3b, uses the centre state of the 3-state HMMs as a speech class from which the within and between class covariances are extracted. Finally in Figure 3c, single-state models are trained and form the speech class.

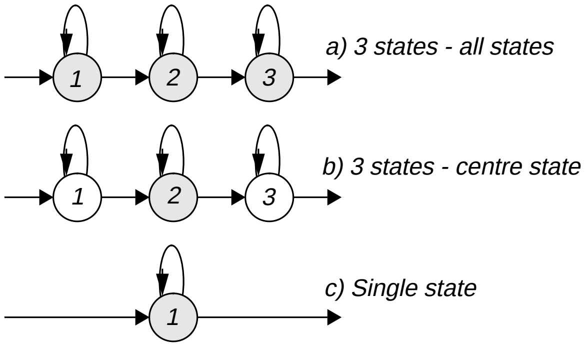

Figure 3: Choice of speech class in LDA - using phoneme models.

3. EXPERIMENTAL RESULTS

This section presents results showing how well various columns and stack widths of the cepstral-time matrix perform for different recognition tasks. The combination of LDA and cepstral-time matrices is shown in the second half of the results section.

In all the experiments, telephony speech is used with an average signal to noise ratio (SNR) of about 35dB although the local SNR is highly variable. The speech is windowed into 32ms frames at a frame rate of 62.5Hz. The frames are transformed into 19-D MFCCs which are then truncated down to 9-D - *MFCC(0)* to *MFCC(8)* - for the MFCC speech feature.

3.1. Effect of stack width and columns

As described in section 2, the cepstral-time matrix is formed by applying a temporal DCT to a stack of MFCC speech vectors. This section presents results for varying the number of MFCC vectors included in the stack (from 3 to 13 frames). Also presented in the results is the effect of using increasing numbers of columns of the cepstral-time matrix to form the final speech feature. In each case column *zero* is always removed, with results showing performance for using just column *one* up to using columns *one* to *six* - where possible. Clearly the maximum number of columns available in the cepstral-time matrix is limited by the stack width. Tables 2 to 4 show recognition accuracy for the selection of stack widths and columns using three different speech databases - UK isolated digits, UK connected digits and a sub-word based town names test.

The UK isolated and connected digit vocabularies consist of the digits *one, two, ..., nine, nought, oh* and *zero.* These are trained on about 500 digit strings. The digits are modeled using 6-state, 7-mode diagonal covariance HMMs with no skip states. Isolated digit recognition performance is quoted in terms of digit accuracy, while for the connected digits, recognition is quoted in terms of string accuracy.

For the town names experiments, 43 monophone models are trained using 3000 phonetically rich spoken sentences from the BT Subscriber database [6]. A grammar is then used to specify 156 English town names from these monophone models. Each monophone is modeled using a 3-state, 12-mode diagonal covariance HMMs.

Width	1	1-2	1-3	1-4	1-5	1-6
3	92.5	92.9	X	X	X	X
5	95.4	96.1	95.7	94.8	X	X
7	95.2	96.7	96.9	97.4	96.8	96.9
9	94.8	96.8	97.6	97.2	96.1	96.9
11	94.7	97.2	97.0	97.6	97.3	97.2
13	94.7	97.1	96.9	97.0	97.0	97.1

Table 2: UK Isolated digit recognition performance - digit accuracy

Width	1-2	1-3	1-4	1-5
3	51.3	X	X	X
5	54.9	57.6	50.7	X
7	56.4	60.5	59.9	57.8
9	58.9	61.8	60.3	60.1
11	55.6	63.1	57.6	58.7
13	51.2	57.4	58.3	58.9

Table 3: UK Connected digit recognition performance - string accuracy.

Width	1-2	1-3	1-4
3	61.8	X	X
5	67.8	69.4	70.7
7	64.2	69.9	70.7
9	61.8	68.8	71.0
11	56.1	66.7	68.0

Table 4: Sub-word town names recognition performance.

The results from tables 2 to 4 show that best performance is typically obtained using the first 3 or 4 columns of the cepstral-time matrix. For the isolated digits good performance is still obtained with just the first 2 columns. For the sub-word based town names, performance increases as more columns are included - this indicates that the faster moving temporal information, stored in these columns, is useful in discriminating between phonemes.

Stack widths of between 7 and 11 frames give best performance for both isolated and connected digits. Best connected digit performance of 63.1% is given with an 11 frame stack and using columns 1 to 3. The results for sub-word based town names indicate that including larger numbers of columns gives better performance.

For comparative purposes Table 5 shows the performance for the best combination of stack widths and column selection obtained for each of the three recognition tasks. The best result achieved using a feature comprising 9 static and 9 velocity MFCCs is also displayed.

Task	MFCCs	Best CTM
UK Isolated Digits	93.7	97.6
UK Connected Digits	55.3	63.1
Sub-word Town names	66.4	71.0

Table 5: Best performance using MFCC+differentials.

These results clearly show that for each of the four tasks the performance of the cepstral-time matrix has outperformed that attained with an MFCCs based feature. Best performance increases are achieved for the two digit tasks.

3.2. Cepstral-time matrices and LDA

Two different experiments combining LDA and cepstral-time matrices are described.

Using the isolated digits database a set of full covariance HMMs are trained using a 9x7 CTM feature produced from a stacking of 7 9-D MFCCs. A second set of HMMs are trained using a feature comprising a stacking of 7 9-D HMMs. The difference between the two feature being the temporal DCT. LDA transform matrices are computed from both sets of HMMs and the two features transformed into their new LDA derived feature space. The feature produced by combining LDA and cepstral-time matrices is refered to as LDA-CTM and the LDA-stacked MFCC combination refered to as LDA-stacked. For both tests dimensions are successively removed from the feature vector and recognition accuracy recomputed. Figure 4 shows recognition performance for the two features.

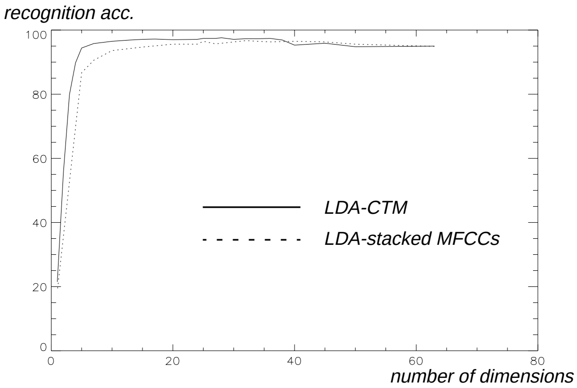

Figure 4: Recognition performance for two feature types following LDA and truncation.

At higher dimensions the performance difference between the two features is small. As more dimensions are removed performance of the LDA-CTM feature remains consistantly higher than the LDA-stacked feature. This is particularly noticable below about 20 dimensions. The LDA-CTM feature is able to sustain higher accuracy as the dimensionality reduces - performance is still above 95% with only 5 dimensions. This represents a drop in accuracy of only 2.6% compared to the best cepstral-time matrix configuration shown in table 1 - stack width 9 with columns 1, 2 and 3 resulting in a 27-D feature. These results show that using a DCT to transform the MFCC stack gives better performance than allowing LDA to determine the transform.

The second experiment examines the 3 different implementations of LDA shown in figure 3. In this experiment the 156 town names task using 3 state monophone HMMs is used. The base feature to which LDA is applied is the 9x3 cepstral-time matrix produced from a stacking of 7 MFCC frames. From table 4 this feature has a baseline accuracy of 69.9%. Figure 5 shows recognition performance for the three different LDA implementations as a function of the dimensionality. The dash-dot line indicates the baseline performance of the 9x3 feature.

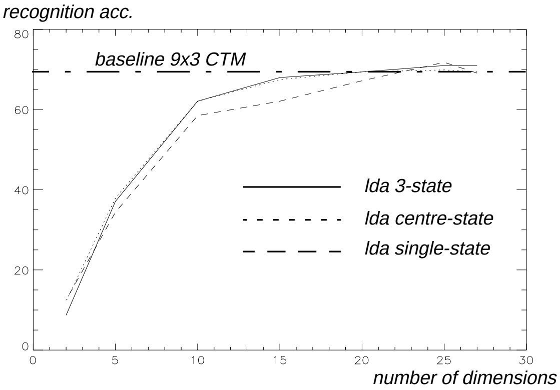

Figure 5: Three implementations of LDA for the sub-word town names task.

Of the three implementations of LDA, best overall performance of 71.9% is given by the single-state method. This is achieved using 25 dimensions and is an increase of 2% over the baseline performance of the untransformed 9x3 cepstral-time matrix. Below 23 dimension the single-state LDA performance falls below that of the other two methods. With 20 dimensions recognition performance for both the 3-state and centre-state based LDA methods is equal to the baseline of 69.9%, giving a dimensionality reduction of 35%.

Although the baseline features are different sizes, figures 4 and 5 show that LDA is able to compress discriminitive information into a smaller number of dimensions for the isolated digit task than for the sub-word task. It is possible this a direct result of the higher feature space complexity associated with the sub-words.

4. CONCLUSION

The main result of this paper is that the previously reported performance improvements which cepstral-time matrices have achieved over MFCCs augmented with higher order derivatives on isolated words can be successfully extended to connected digits and a monophone-based town names test. Results have shown that slightly different truncations of the matrix give best performance for the different tasks. Isolated digits achieve good performance using a little as 2 columns, whereas connected digits typically require between 3 and 4 columns. Monophone-based tests require about 4 columns. This indicates that the faster moving temporal information is supplying useful discriminative information to the HMMs.

LDA is also shown to give an improvement in recognition accuracy as well as reducing the size of the final speech feature. In particular LDA is shown to be very effective for the simple isolated digit task and less effective for the more complex sub-word task.

5. REFERENCES

1. B.P. Milner, “Inclusion of temporal information into features for speech recognition”, Proc. ICSLP, pp. 256-259, 1996.
2. 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.
3. B.A. Hanson and T.H. Applebaum, “Robust speaker-independent word features using static, dynamic and acceleration features; experiments with Lombard and noisy speech”, Proc. ICASSP, pp. 857-860, 1990.
4. E.S. Parris and M.J. Carey, “Estimating linear discriminant parameters for continuous density hidden markov models”, Proc. ICSLP, pp. 215-218, 1994.
5. T. Parsons, Voice and Speech Processing, McGraw-Hill, 1987.
6. A.D. Simons and K. Edwards, “Subscriber - A phonetically annotated telephony database”, Proc. IOA, Vol. 14, Pt. 6, pp. 3-15, 1992.