A COMPARISON OF FRONT-END CONFIGURATIONS FOR ROBUST SPEECH RECOGNITION

Ben Milner

School of Information Systems, University of East Anglia, Norwich, UK
bpm@sys.uea.ac.uk

ABSTRACT

This paper presents a comparative analysis of the processing stages involved in feature extraction for speech recognition. Feature extraction is considered as comprising three different processing stages; namely static feature extraction, normalisation and inclusion of temporal information. In each stage a comparison of techniques is made, both theoretically and in terms of their comparative performance.

The analysis shows that while some techniques may appear significantly different, upon analysis the effect they have on the signal can be similar. Comparative studies include MFCC and PLP analysis, RASTA filtering and cepstral mean normalisation, and temporal derivatives and cepstral-time matrices. Experimental results, on an unconstrained monophone task, compare recognition performance using different front-end configurations.

1. INTRODUCTION

A significant amount of effort has been devoted to establishing speech feature extraction schemes which enable robust and high performance speech recognition in a range of operating environments [1,2]. To this effect a considerable number of alternative processing schemes have been proposed. The aim of this work is to compare, both theoretically and experimentally, a number of the more popular techniques and identify which combinations work best.

The analysis is made by considering feature extraction as comprising a number of independent processing stages. The first stage involves extracting an instantaneous, or static, compressed representation of a short duration window of the speech signal. Section 2 analyses two such methods, namely mel-frequency cepstral coefficients (MFCC) and perceptual linear prediction (PLP) cepstra. The purpose of the second stage is to normalize the static parameters. Techniques for this are compared in section 3. The final stage involves the computation and inclusion of temporal information into the speech feature - section 4 reviews this. Following the theoretical comparison of the three processing stages, section 5 presents experimental results in which a comparison of the performance of different front-end configurations is made. Finally a conclusion is made in section 6.

2. STATIC FEATURES

A speech signal is formed by convolving the excitation signal from the lungs with the filter response of the vocal tract. For speech recognition, the vocal tract component provides best discrimination between speech sounds. Most feature extraction methods use cepstral analysis to extract this vocal tract component from the speech signal. A number of procedures have been proposed for computing cepstral features, however the most successful of these also include attributes of the psychophysical processes of human hearing into the analysis. This section compares two such methods; MFCC analysis and PLP analysis.

2.1. MFCC and PLP Analysis

MFCC analysis has been well reported [1] and utilises a mel-filterbank which is designed to model the hair spacings along the basilar membrane of the ear. The right-hand side column of figure 1 illustrates the extraction of static MFCCs from a speech signal.

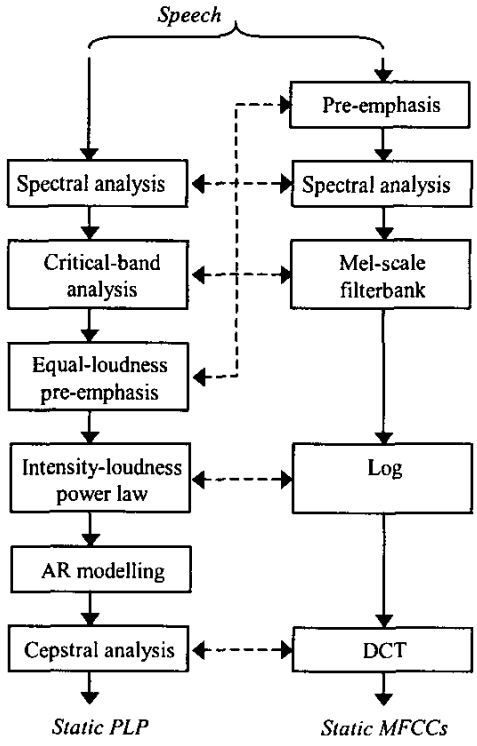

Figure 1: Comparison of PLP and MFCC analysis

Hermansky [2] formulated PLP as a method for deriving a more auditory-like spectrum based on linear predictive (LP) analysis of speech. Conventional LP analysis approximates the high

0-7803-7402-9/02/$17.00 ©2002 IEEE

energy areas of the spectrum (formants) and smoothes out the finer harmonic structure. This estimation is done equally well at all frequencies which is inconsistent with human hearing. The auditory-like spectrum in PLP is achieved by making some engineering approximations of the psychophysical attributes of the human hearing process. The left-hand side column of figure 1 illustrates the process for deriving the auditory spectrum from which all-pole modelling and cepstral analysis is performed.

2.2. Comparison of MFCC and PLP Analysis

The broken arrows in figure 1 link processing stages of MFCC and PLP analysis which are similar. The remainder of this section considers each of these processing stages in more detail.

Spectral analysis – PLP and MFCC analysis both obtain a short-term power spectrum by applying a Fourier transform to a frame of Hamming windowed speech, typically 20-32ms in duration.

Critical-band analysis – Both PLP and MFCC analysis employ an auditory-based warping of the frequency axis derived from the frequency sensitivity of human hearing. MFCCs are based on a uniform spacing along the mel-scale whereas PLP uses the Bark scale. Figure 2 shows the similarity of the mel and Bark scale warping functions.

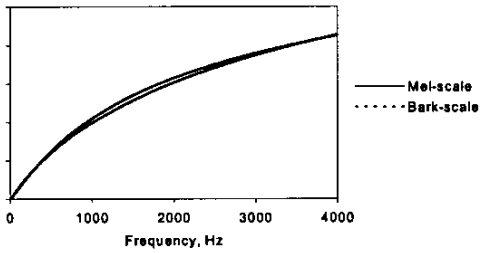

Figure 2: Mel-scale and Bark-scale warping functions

At regular points along the two scales, windowing functions are applied which quantise the frequency spectrum. Mel-scale filterbank analysis uses triangular shaped windows whereas in PLP analysis the window shape is designed to simulate critical-band masking curves. Figure 3 shows the shape of these windows.

Equal-loudness pre-emphasis – To compensate for the unequal sensitivity of human hearing across frequency, PLP analysis scales the critical bands amplitudes according to an equal-loudness pre-emphasis function, such as

$$E(\omega)=\frac{\left(\omega^2+56.8\times10^6\right)\omega^4}{\left(\omega^2+6.3\times10^6\right)^2\left(\omega^2+0.38\times10^9\right)} \quad (1)$$

In MFCC analysis, pre-emphasis is applied in the time-domain. A typical implementation uses a first-order high pass filter

$$H(z)=1-\alpha z^{-1} \quad (2)$$

Both critical band analysis and mel-filterbank analysis can be viewed as applying a set of basis functions to the power spectrum of the speech signal. From the processing stages outlined above, figure 3 shows the similarity of the basis functions used in PLP and MFCC analysis.

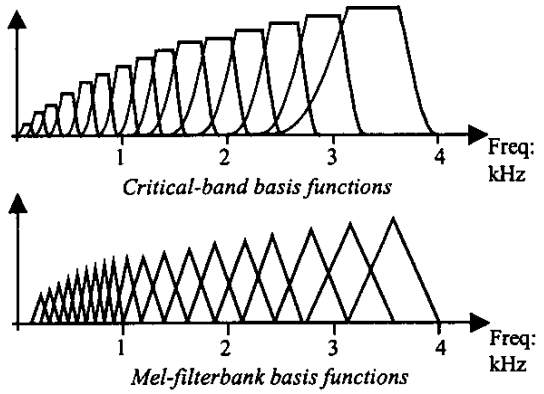

Figure 3: Critical-band and mel-filterbank basis functions

The non-linear spacing of both the PLP and MFCC windows along the frequency axis displays similar characteristics. Both allocate more filters to the lower frequencies where hearing is more sensitive. The amplitude of the critical-band filters and mel-filterbank channels are determined by the pre-emphasis functions. Again considerable similarity is displayed by the two approaches which emphasise higher frequency components. The general shape of the analysis windows also exhibit similar properties. However, critical-band filters are flat topped and non-symmetric with wider skirts on the low frequency side. In contrast mel-filterbank windows are triangular and symmetric.

Intensity-loudness power law – This processing stage models the non-linear relation between the intensity of sound and its perceived loudness. Cubic root compression of critical-band energies is used to implement this function in PLP. In MFCC analysis, logarithmic compression of the mel-filterbank channels is applied. Again the result of these two operations has a very similar effect.

AR Modelling and Cepstral Analysis – It is at this stage that PLP and MFCC analysis diverge. MFCC analysis computes cepstral coefficients from the log mel-filterbank using a discrete cosine transform. However in PLP analysis the critical-band spectrum is converted into a small number of LP coefficients through the application of an inverse DFT to provide autocorrelation coefficients. From the LP coefficients, cepstral coefficients are computed and these form the final static feature vector.

3. NORMALISATION METHODS

Following computation of static features, front-end processing techniques typically employ some form of normalisation to the feature vector stream. In this comparison the processes of RASTA filtering and cepstral-mean normalisation are considered.

3.1. RASTA Filtering

The RASTA filter was proposed by Hermansky [3] as a front-end operation to reduce both communication channel effects and noise distortion. The RASTA filter is implemented as a 4th order IIR bandpass filter which is applied to the time series of static

feature vectors. Channel distortion is additive in the log frequency and cepstral domains, so applying a sharp cut-off highpass filter to each coefficient, over time, removes the offset and hence suppresses the channel distortion. Several improvements have been proposed, such as J-RASTA, phase corrected RASTA, and the automatic computation of the filters [4].

3.2. Cepstral Mean Normalisation (CMN)

Cepstral mean normalisation (CMN), or subtraction (CMS), [5] assumes that most channel distortions are stationary (such as a microphone), or at least slowly time-varying. This means that they impart a near constant offset, over time, on each separate log filterbank or cepstral coefficient. Calculating the mean of each coefficient across a reasonably large number of frames gives the cepstral mean. Subtracting this from the original cepstral vectors removes channel induced offsets together with any other stationary speech components.

3.3. Comparison of RASTA and CMN

The bandpass nature of the RASTA filter and mean subtraction of CMN both result in a feature vector stream with mean of zero. The RASTA filter implementation is more straightforward than CMN which imparts a significant delay while computing the cepstral mean. Improvement, such as dynamic CMN, partially solve this by using an IIR filter to store a running cepstral average. However this makes the CMN implementation even more similar to the RASTA filter.

4. TEMPORAL INFORMATION

The majority of classifiers used for speech recognition are based on hidden Markov models (HMMs). An number of assumptions need to be made when using HMMs to model speech; one of which is the assumption that the observation vectors are generated from an independent identically distributed (IID) process. The temporal correlation which exists in the feature vector stream, however, breaks this assumption. Including temporal information into speech features partly reduces the effect of violating the IID assumption. A number of studies [6,7] have shown the importance of including temporal information into speech features. This section compares the conventional method of extracting temporal derivatives with the cepstral-time matrix approach of encoding temporal information.

4.1. Temporal Derivatives

The inclusion of temporal derivatives, or velocity and acceleration components, into speech feature vectors has been very well documented [6]. Velocity is typically calculated using a simple difference or regression over a window of five static vectors. Acceleration is usually computed as a simple difference over a window spanning three velocity vectors.

4.2. Cepstral-time matrices

The cepstral-time matrix (CTM) is an alternative framework for encoding the temporal variations of speech into the feature [7]. Figure 4 illustrates the computation of temporal information by applying a discrete cosine transform (DCT) across a window of typically 7 cepstral vectors and an energy term. The columns of the resulting matrix represents different temporal regions and can be truncated according to the amount of temporal information required in the final speech feature.

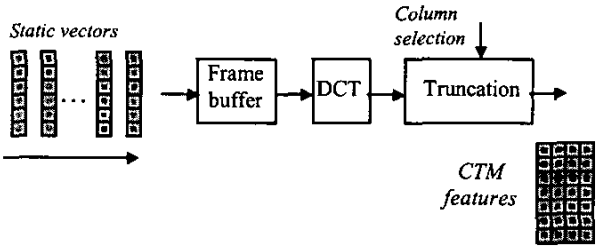

Figure 4: Generation of the cepstral-time matrix

Matrix truncations suitable for speech recognition typically comprise the first few columns of the CTM.

4.3. Comparison of Temporal Derivatives and CTM

As has been reported in [7], figure 5 illustrates the similarity of the basis functions used for computing temporal derivatives (solid lines) and the columns of the CTM (broken lines).

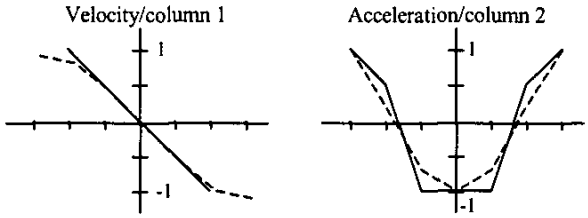

Figure 5: Basis functions for temporal derivatives and CTM

The basis function for velocity is shown to be a ramp where as the CTM column 1 basis function is a half cosine wave. Similarly, the acceleration basis function is parabolic while that for column 2 of the CTM is a whole cosine wave. It is interesting to observe that the CTM basis functions are of equal length, whereas the number of frames used to compute temporal derivatives increase for higher derivatives.

5. EXPERIMENTAL RESULTS

To constrain the experimental results so only the effect of feature extraction is considered, tests have been performed on an unconstrained monophone task. These have been based on the BT Subscriber telephony database which contains approximately 4330 sentences in the training set and 2560 in the test set. Each of the 44 phonemes is modelled by a 3-state, 12-mode, diagonal covariance HMM. The grammar allows completely unconstrained phoneme recognition with optional noise between phonemes. In the feature extraction schemes a frame rate of 16ms has been used, together with a Hamming window width of 32ms.

A number of different front-end configurations have been used in the experiments. Figure 6 illustrates the various front-end configurations which are possible by connecting together different processing stages. Static features are computed from either MFCC or PLP analysis. Normalisation takes the form of RASTA filtering or CMN or no normalisation. Temporal information is represented through either time derivatives or CTM processing.

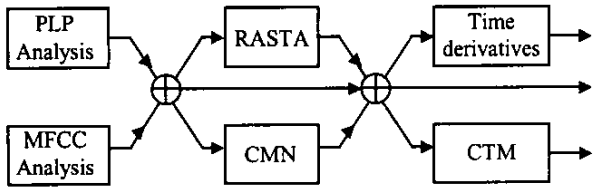

Figure 6: Selection of final speech feature

Experimental results are shown in table 1, where the ⊕ operator indicates the inclusion of the various processing stages. Both MFCC and PLP analysis are configured to output an 8-D cepstral vector and log energy term.

Parameterisation	*Accuracy, %*
1. MFCC ⊕ VEL ⊕ ACC	37.1
2. MFCC ⊕ CMN ⊕ VEL ⊕ ACC	39.7
3. MFCC ⊕ RASTA ⊕ VEL ⊕ ACC	39.9
4. MFCC ⊕ CTM(1,2,3)	33.3
5. MFCC ⊕ RASTA ⊕ CTM(1,2,3)	36.8
6. MFCC ⊕ CTM(0,1,2,3)	38.2
7. MFCC ⊕ RASTA ⊕ CTM(0,1,2,3)	45.5
8. PLP ⊕ CTM(1,2,3)	30.8
9. PLP ⊕ RASTA ⊕ CTM(1,2,3)	33.0
10. PLP ⊕ RASTA ⊕ CTM(0,1,2,3)	40.1

Table 1: Monophone accuracy for various parameterisations

The first three rows of the table consider the effect of normalisation in the front-end. In each case the speech feature comprises a static MFCC vector with velocity and acceleration components. With no normalisation monophone accuracy is 37.1%, but applying CMN increases this to 39.7%. Normalising the MFCCs with a RASTA filter improves performance slightly to 39.9%.

Rows 4 to 7 show the effect of using the cepstral-time matrix for representing temporal information. Using columns 1,2 and 3 of the cepstral-time matrix as the speech feature results in an almost 4% (absolute) reduction in performance in comparison to static, velocity and acceleration MFCCs (row 1). Including RASTA filtering in the CTM configuration improves performance by 3.5%. This also reduces the performance gap, in comparison to row 3 of the table, to 3%. Extending the cepstral-time matrix to also include the zeroth column (static information) in addition to RASTA filtering, increases performance substantially to 45.5%. This is considerably higher than that attained with the comparable RASTA filtered MFCC with velocity and acceleration terms.

The effect of substituting PLP analysis for MFCC analysis is shown in the last three rows of the table. Using PLP with the three column CTM, results in a 2.5% reduction of performance in comparison with MFCC analysis. Including the zeroth column of the CTM, together with RASTA filtering, increases performance to 40.1%. This is again lower than comparable MFCC-based analysis.

6. CONCLUSIONS

This work has considered feature extraction as a three-stage system comprising static feature analysis, normalisation and the inclusion of temporal information. Within each of these stages a comparison has been made of methods which are commonly used.

Static feature extraction has considered both MFCC and PLP cepstral analysis. A comparison of the frequency compression shows that critical band analysis in PLP and mel-filterbank analysis used in MFCCs are remarkably similar. Significant differences, however, occur in the remaining processing leading to the cepstral representation. Comparisons on an unconstrained monophone test showed MFCC analysis to give better performance than the PLP derived cepstra. RASTA filtering and cepstral mean subtraction are examined as methods for normalisation. Analysis of these shows that they both remove steady state information from the feature vector stream. Experiments show that RASTA filtering results in slightly better performance on the unconstrained monophone task than CMN. RASTA filtering also provides a better solution for real-time operation as there is no delay in computing the cepstral mean. For temporal modeling the inclusion of temporal derivatives is compared with cepstral-time matrices. Again, analysis of these shows considerable similarity between the techniques. Results show that for non-normalised features, temporal derivates give better performance. However when RASTA filtering is also applied, the cepstral-time matrix gives significantly better performance. Overall best performance was given by RASTA filtered static MFCCs transformed into a cepstral-time matrix as the final speech feature for recognition.

7. REFERENCES

[1] S.B. Davis and P. Mermelstain, "Comparison of parametric representations for monosyllabic word recognition", IEEE Trans. ASSP, vol. 28, no. 4, pp 357-366, 1990.

[2] H. Hermansky, "Perceptual linear prediction (PLP) analysis of speech", Proc. JASA., pp. 1738-1752, 1990.

[3] H. Hermansky and N. Morgan, "RASTA processing of speech", IEEE Trans. Speech and Audio Proc., vol. 2, no. 4, pp. 578-589, October 1994.

[4] J. Koehler, N. Morgan, H. Hermansky, H. G. Hirsch, and G. Tong, "Integrating RASTA-PLP into speech recognition", Proc. ICASSP, pages 421-424, 1994.

[5] S. Furui, "Cepstral analysis techniques for automatic speaker verification", IEEE Trans. ASSP, vol 29, pp. 254-272, 1981.

[6] B.A. Hanson and T.H. Applebaum, "Robust speaker-independent word features using static, dynamic and acceleration features", Proc. ICASSP, pp. 857-860, 1990.

[7] B.P. Milner, "Inclusion of temporal information into features for speech recognition", Proc. ICSLP pp. 256-259, 1996.