A COMPARISON OF FRONT-END CONFIGURATION FOR ROBUST SPEECH RECOGNITION

Author : Ben Milner

Professor:陳嘉平
Reporter:葉佳璋

Outline

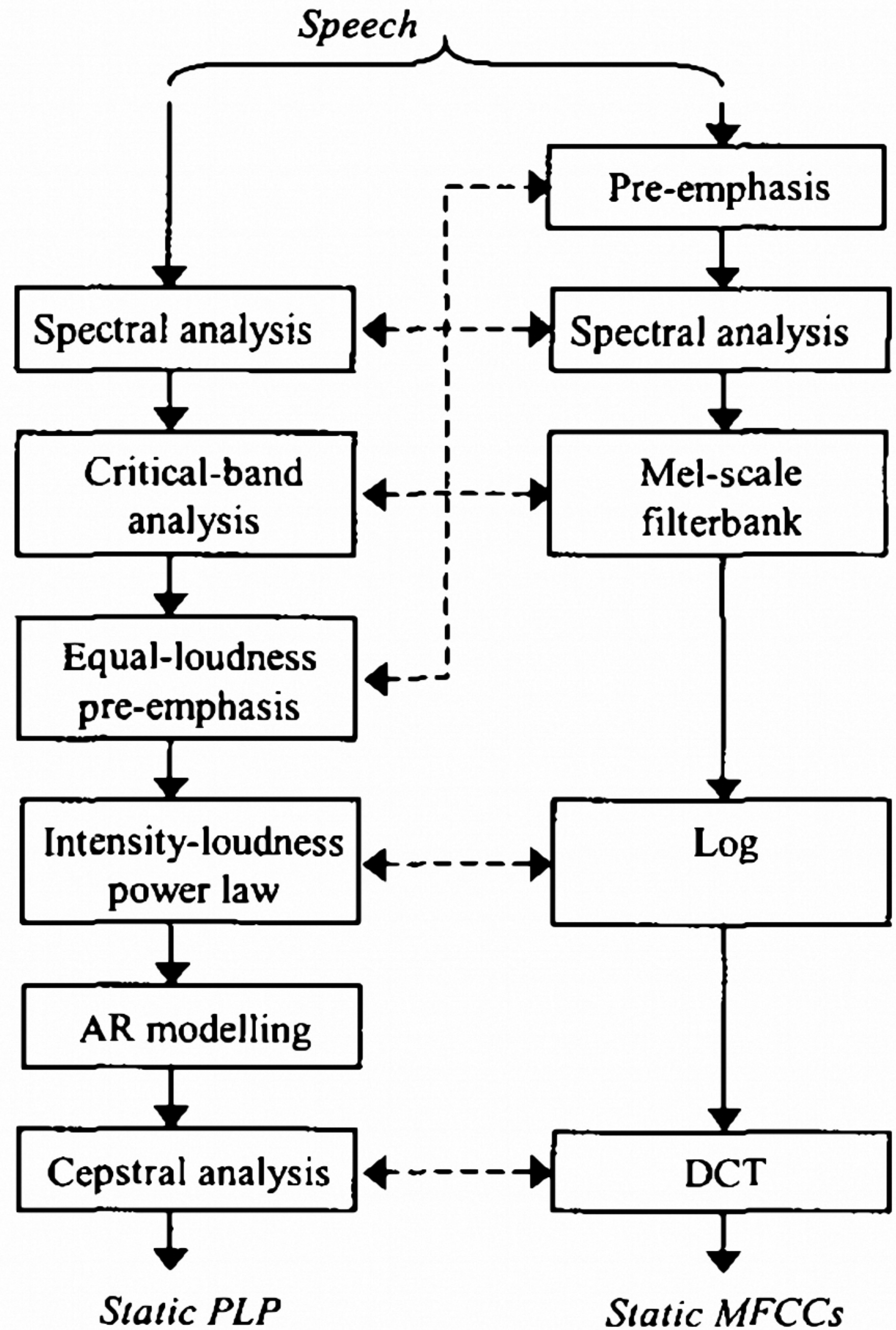
- Introduction
- Static features
- Normalization Methods
- Temporal Information
- Experimental Results

Introduction

- Feature extraction is considered as comprising three different processing stage; namely static feature extraction, normalization and inclusion of temporal information.
- The aim of this work is to compare, both theoretically and experimentally, a number of more popular techniques and identify which combinations work best.

Static features extraction

- For speech recognition, the vocal tract component provides best discrimination between speech sounds.
- Most of feature extraction methods use cepstral analysis to extract this vocal tract component from computing cepstral feature.
- Successful of these methods also include attribute of the psychophysical processes of hearing into the analysis.


Speech
Pre-emphasis
Spectral analysis
Spectral analysis
Critical-band analysis
Mel-scale filterbank
Equal-loudness pre-emphasis
Intensity-loudness power law
Log
AR modelling
Cepstral analysis
DCT
Static PLP
Static MFCCs

Comparison of MFCC and PLP Analysis

- Spectral analysis:

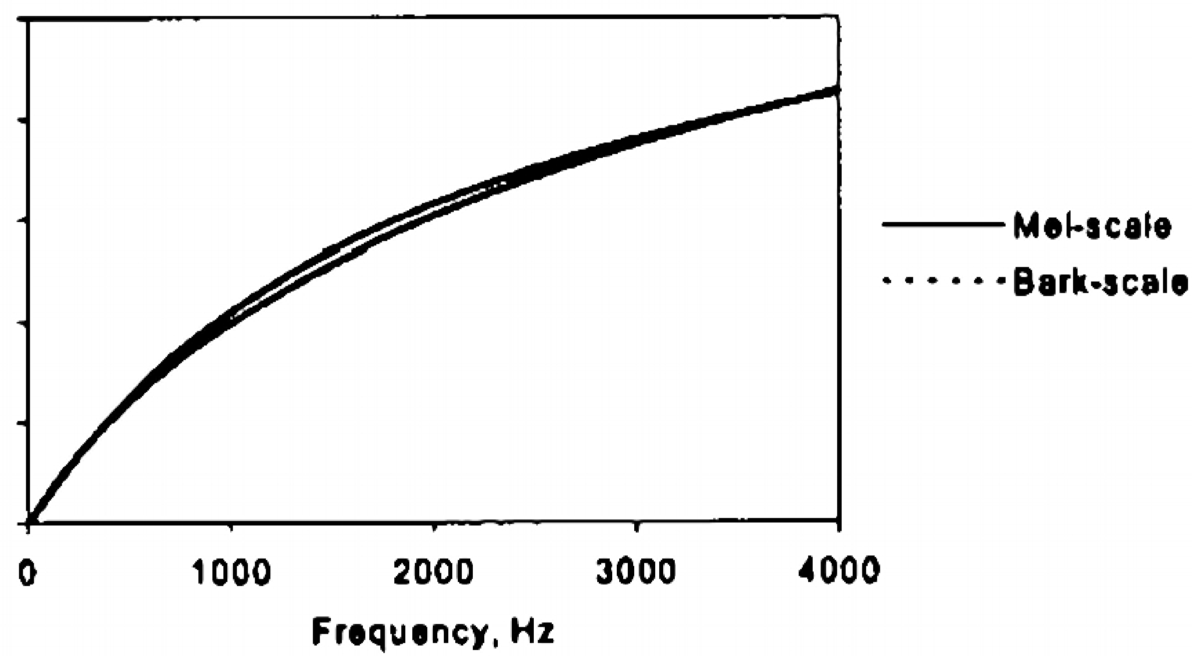
  PLP and MFCC analysis both obtain a short-term power spectrum by applying a Fourier transform to a frame of Hamming windowed speech, typically 20-32ms in duration.

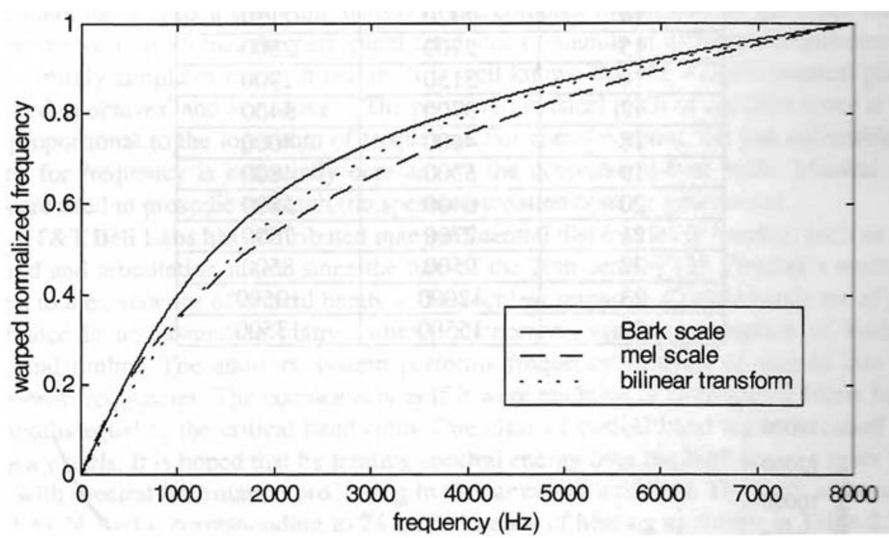
Comparison of MFCC and PLP Analysis(cont.)

- Critical-band analysis:

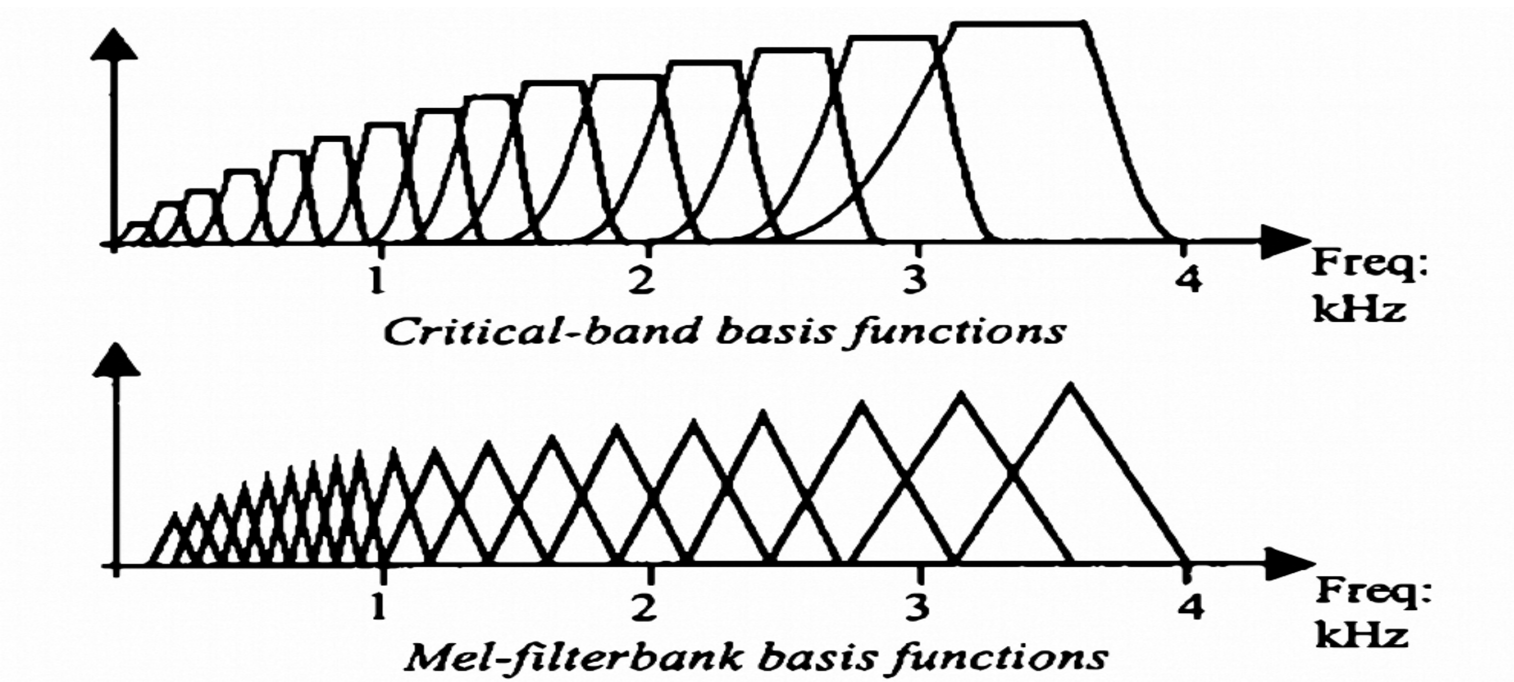
  Both PLP and MFCC employ an auditory-based warping of the frequency axis derived from the frequency sensitivity of human hearing.

  MFCC are based on a uniform spacing along the Mel-scale and PLP uses the Bark scale.


Mel-scale
Bark-scale
0
1000
2000
3000
4000
Frequency, Hz


warped normalized frequency
0
0.2
0.4
0.6
0.8
1
0
1000
2000
3000
4000
5000
6000
7000
8000
frequency (Hz)
Bark scale
mel scale
bilinear transform

- Both critical band analysis and Mel-filter analysis can be view as applying a set of basis function to the power spectrum of the speech signal.



Comparison of MFCC and PLP Analysis(cont.)

- 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

$$E(w) = \frac{(w^2 + 56.8 \times 10^6) w^4}{(w^2 + 6.3 \times 10^6)^2 (w^2 + 0.38 \times 10^9)}$$

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

$$H(z) = 1 - az^{-1}$$

Comparison of MFCC and PLP Analysis(cont.)

- Intensity-loudness power law:

  This processing stage models the non-linear relation between the intensity of sound and its perceived loudness.

  In the PLP, cubic root compression of critical-band energies is used to implement this function.

  In the MFCC analysis, logarithmic compression of the mel-filterbank channels is applied.

Comparison of MFCC and PLP Analysis(cont.)

- AR Modeling and Cepstral Analysis:

  MFCC analysis computes cepstral coefficients from the log Mel-filter using a discrete cosine transform.

  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.

Normalization Methods

- Following computation of static features, front-end processing techniques typically employ some form of normalization to the feature stream.

- In this comparison the process
  RASTA filtering
  cepstral-mean normalization

RASTA Filtering

- The RASTA filter as a front-end operation to reduce both communication channel effects and noise distortion.

- Channel distortion is additive in the log frequency and cepstral domain, so applying a sharp cut-off high pass to each coefficient, over time, remove the offset and suppresses the channel distortion.

Cepstral Mean Normalization(CMN)

- 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 component.

Comparison of RASTA and CMN

- The bandpss nature of RASTA and mean subtraction of CMN result in a feature vector stream with mean of zero.

- RASTA filter implementation is more straightforward than CMN, which impart a significant delay while computing the cepstral mean.

Temporal Information

- HMMs need to assumption the observation vector are generated from an independent identically distributed(IID).

- The temporal correlation which exists in the feature vector stream, brakes this assumption.

- encoding temporal information:
  Temporal Derivatives
  Cepstral-time matrices

Temporal Derivatives

- The first-order temporal derivative(velocity)

$$\partial c_t(n) = \frac{\sum_{k=-k}^{k} k c_{t+k}(n)}{\sum_{k=-k}^{k} k^2}$$

Where $\partial c_t(n)$ is the first time derivative of the n^{th} cepstral coefficient at time frame t, and $c_{t+k}(n)$ is the n^{th} coefficient of the t+k^{th} static cepstral vector. The range is -k to +k is the time span of cepstral vector across which the derivative is calculated.

- In a similar way, the second-order temporal derivative (acceleration) $\partial\partial c_t(n)$ is usually compute as simple difference over velocity vector.

Cepstral-time matrices(CTM)

- The cepstral-time matrix is an alternative framework for encoding the temporal variations of speech into the feature.

- 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.

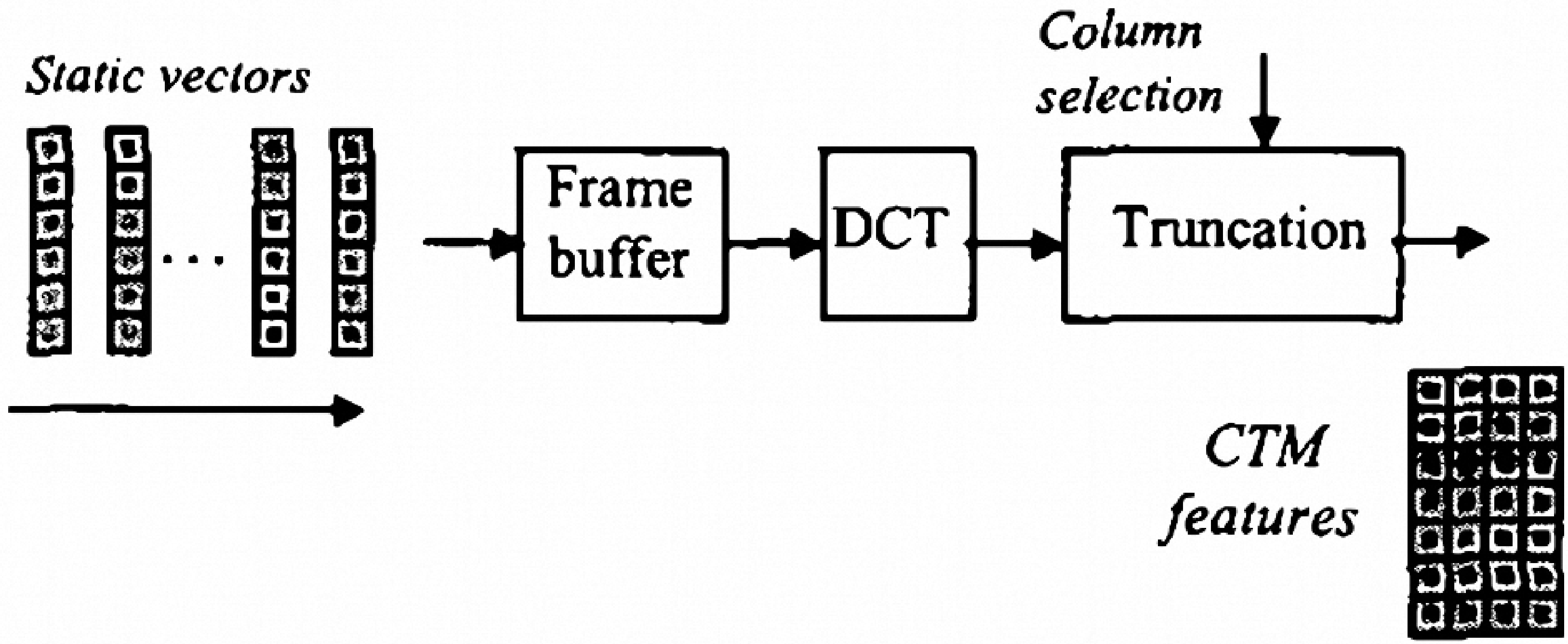
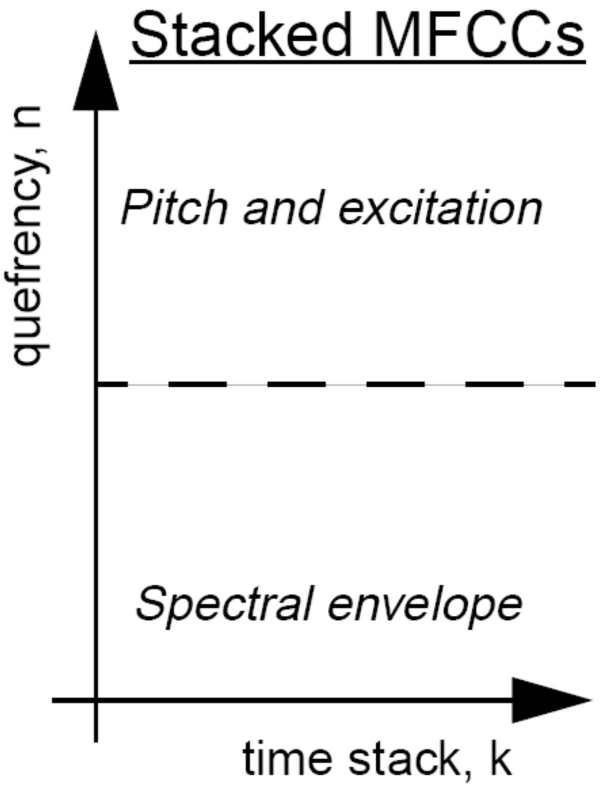
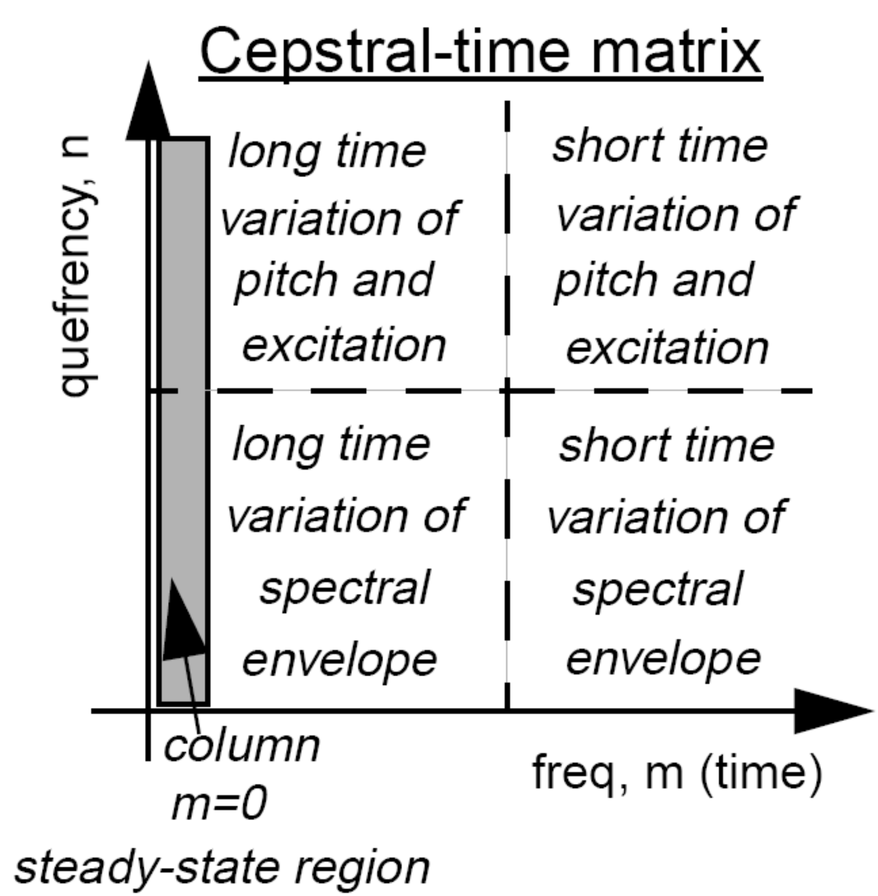


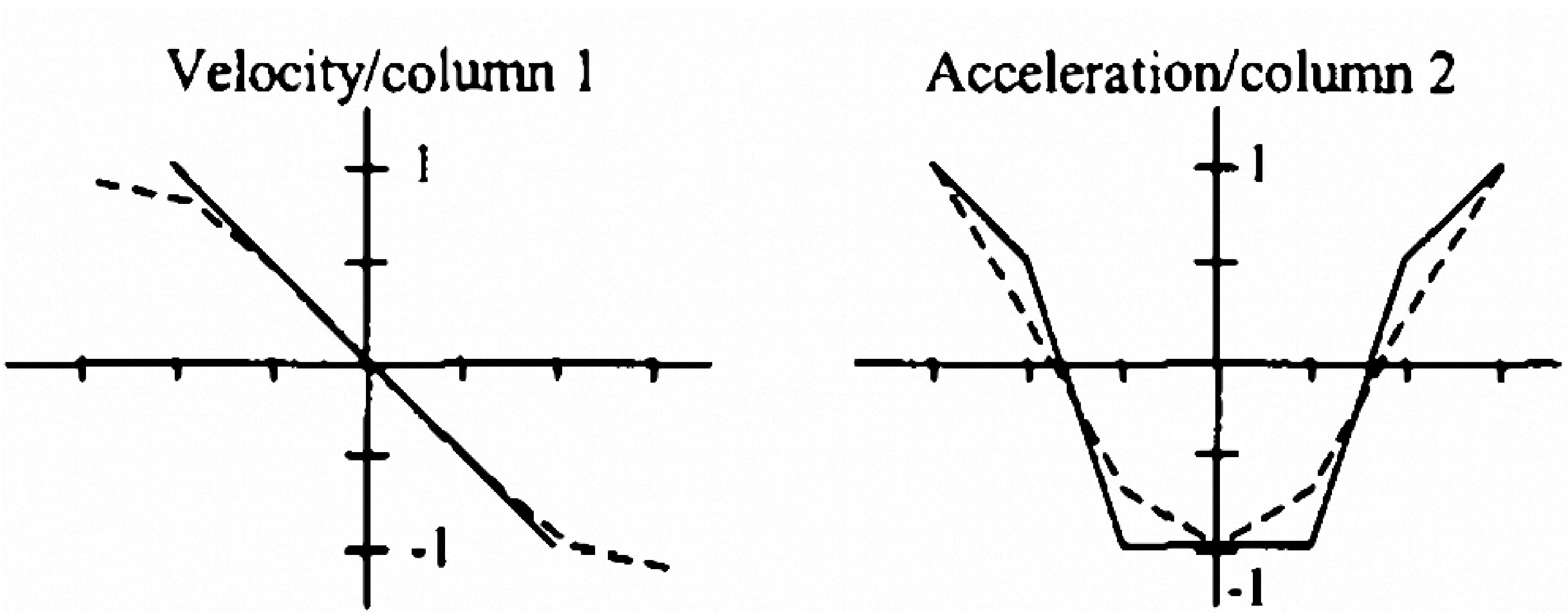
Figure 4: Generation of the cepstral-time matrix

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





Comparison of Temporal Derivatives and CTM



Experimental results

- To constrain the experimental results so only the effect of feature is considered, tests have been performed on an unconstrained monophone task.
- 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 modeled by a 3-state, 12-model, diagonal covariance HMM.
- In the feature extraction schemes a frame rate of 16ms has been used, together with a Hamming window width of 32ms.

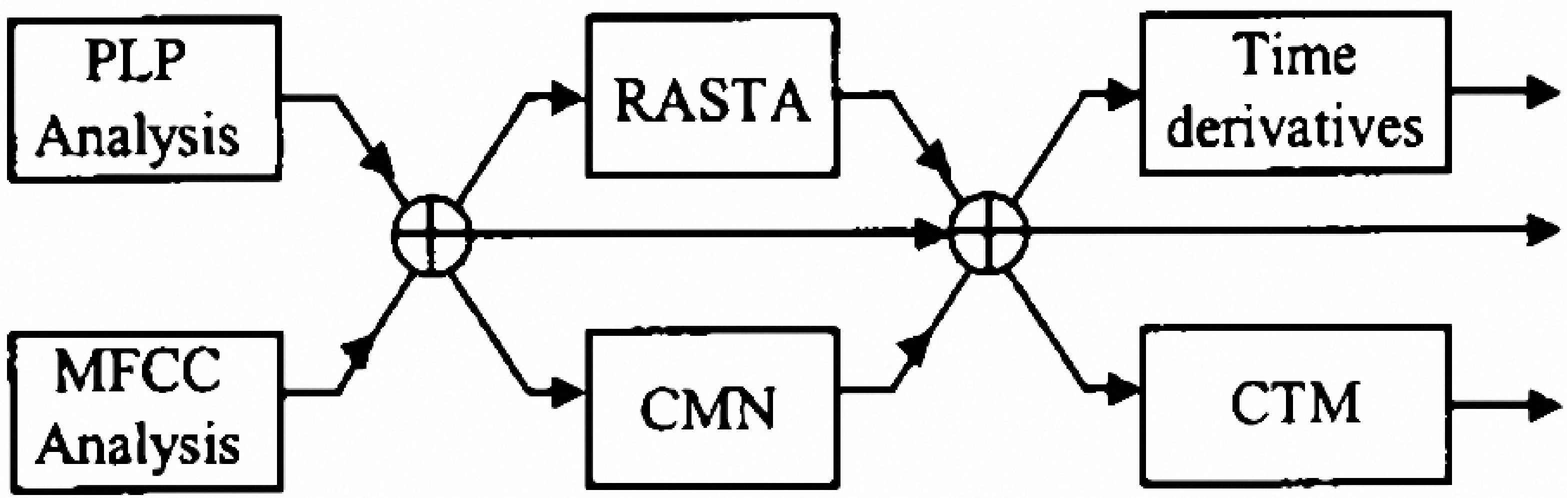


Figure 6: Selection of final speech feature

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