

# *Text Independent Speaker Identification*

CSE 666 Term Project Presentation

Achint Oommen Thomas

Juan Li

# Presentation Overview

---

- Introduction
- Challenges
- Basic Approach
- Pre-processing
- Feature Extraction
- Codebook Generation
- Experimental Results
- References

# Introduction

---

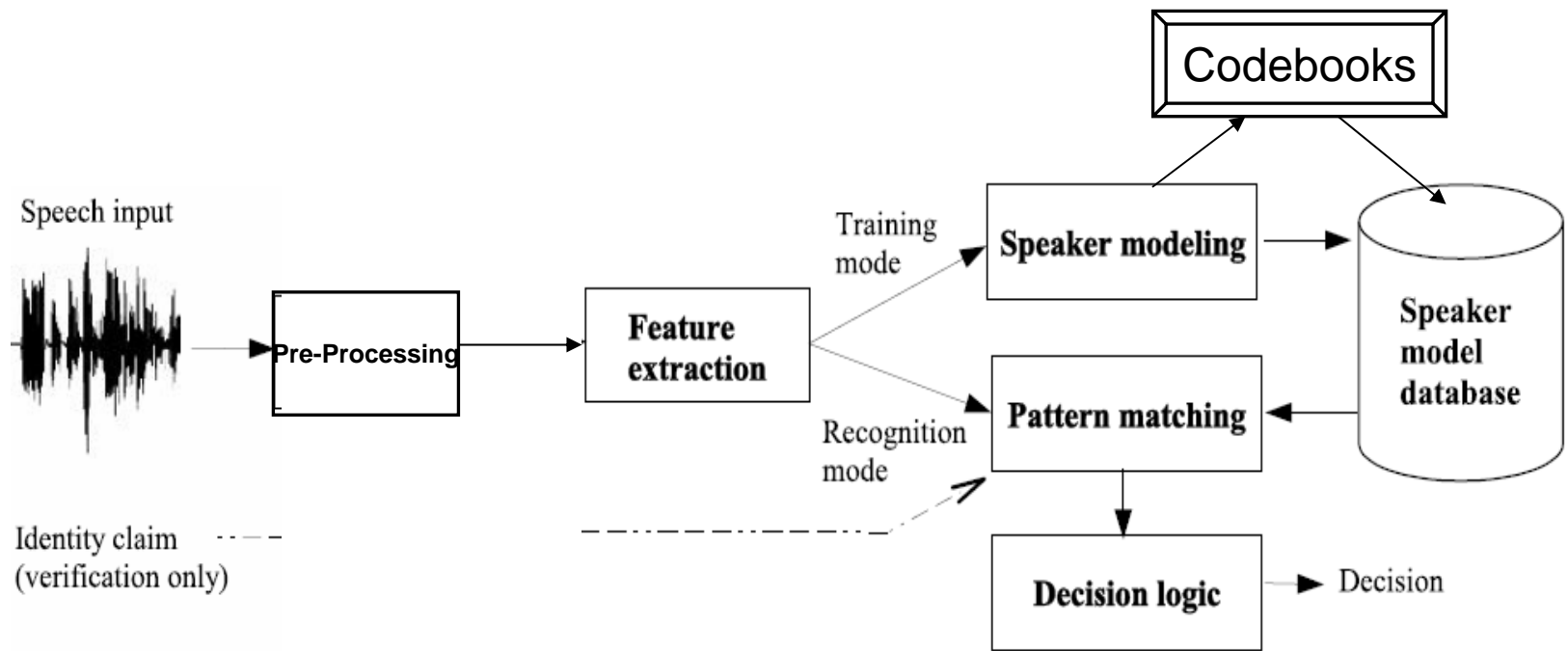
- Speech – Natural, inexpensive
- Speech Recognition vs. Speaker Recognition
- Multi speaker tasks
  - Detection, tracking, segmentation
- Text Dependent vs. Text Independent

# Challenges

---

- Behavioral Biometric
  - Emotional, physical states affect quality
- Can be imitated to a certain degree
- Need to capture discriminating features
- Wide area of expertise needed
  - Speech physiology, acoustic phonetics, digital signal processing, statistical pattern recognition

# Basic Approach



Components of automatic speaker identification system [2]

# Basic Approach – contd.

---

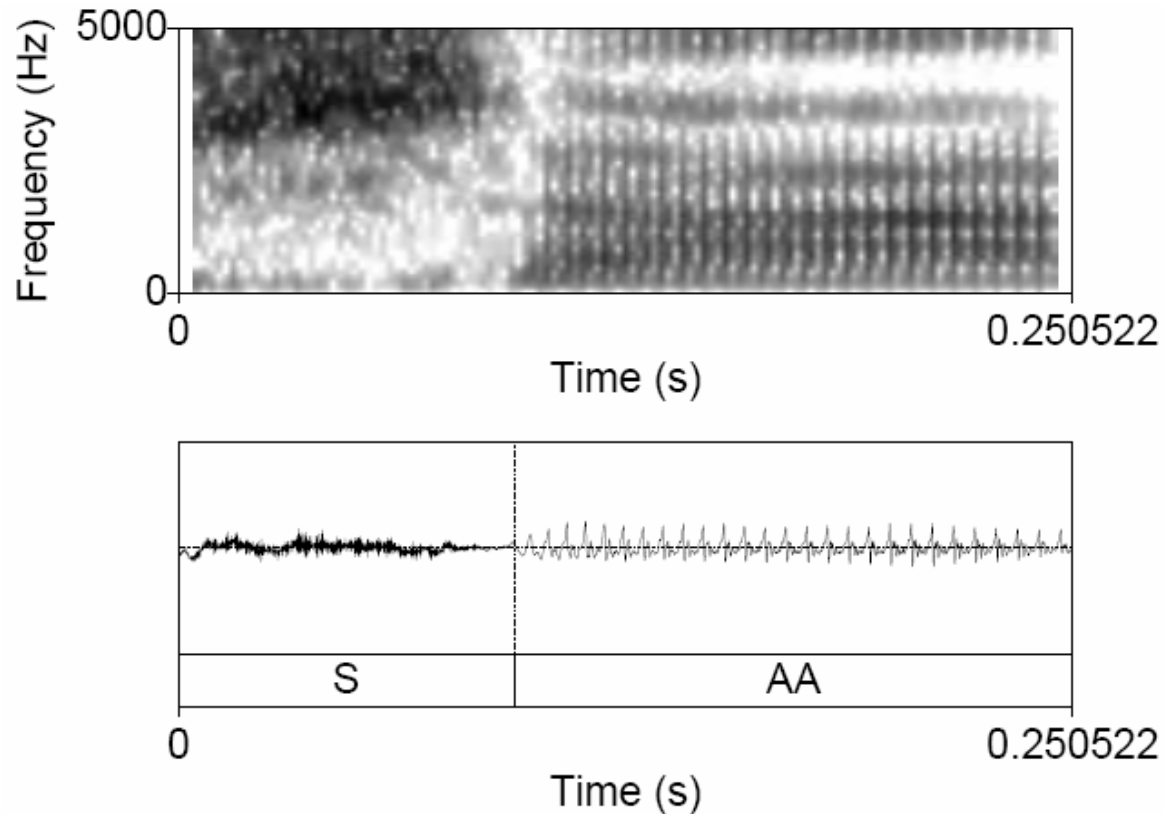
- Pre-Processing
  - Pre-emphasis
- Feature Extraction
  - Mel cepstrum coefficients
- Codebook Generation
  - LBG Algorithm, PCA Variant
- System Testing

# Pre-processing

---

- **Noise Reduction**
  - Environmental noise, Channel noise
- **Silence Removal**
  - Energy based detection methods
- **Pre-emphasis**

# Pre-processing – contd.



Examples of unvoiced and voiced sounds [2]

# Pre-processing – contd.

---

- **Pre-emphasis**
  - Signal processed by a high-emphasis filter
  - Emphasizes higher frequencies as they contain speaker dependent information
  - Voiced glottal sounds have  $-12\text{dB/octave}$  slope
  - Energy radiating at lips gets  $+6\text{dB/octave}$  boost
  - Hence, glottal sounds have  $-6\text{dB/octave}$  net slope
  - Pre-emphasis removes glottal effects from vocal tract parameters

# Pre-processing – contd.

---

- Pre-emphasis – contd.
  - No need to pre-emphasize unvoiced sounds since spectrum is already flat
  - Implemented in time domain as

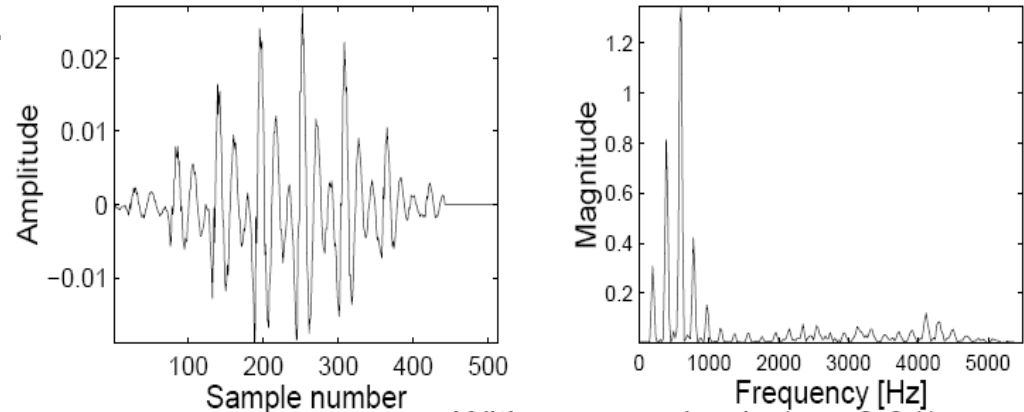
$$s[n] = s[n] - \alpha s[n - 1]$$

- $s[n]$  =  $n^{\text{th}}$  signal sample
- $\alpha$  = slope of filter; usually 0.95

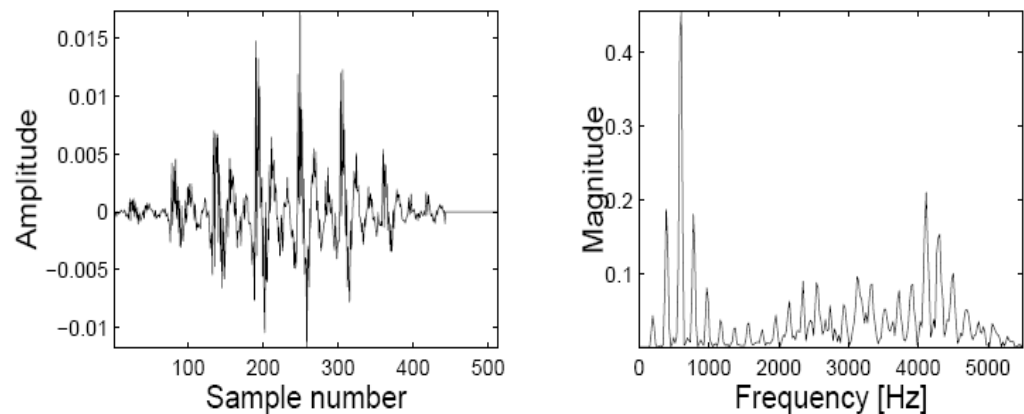
# Pre-processing – contd.

- Pre-emphasis – contd.

Without pre-emphasis



With pre-emphasis ( $\alpha = 0.91$ )



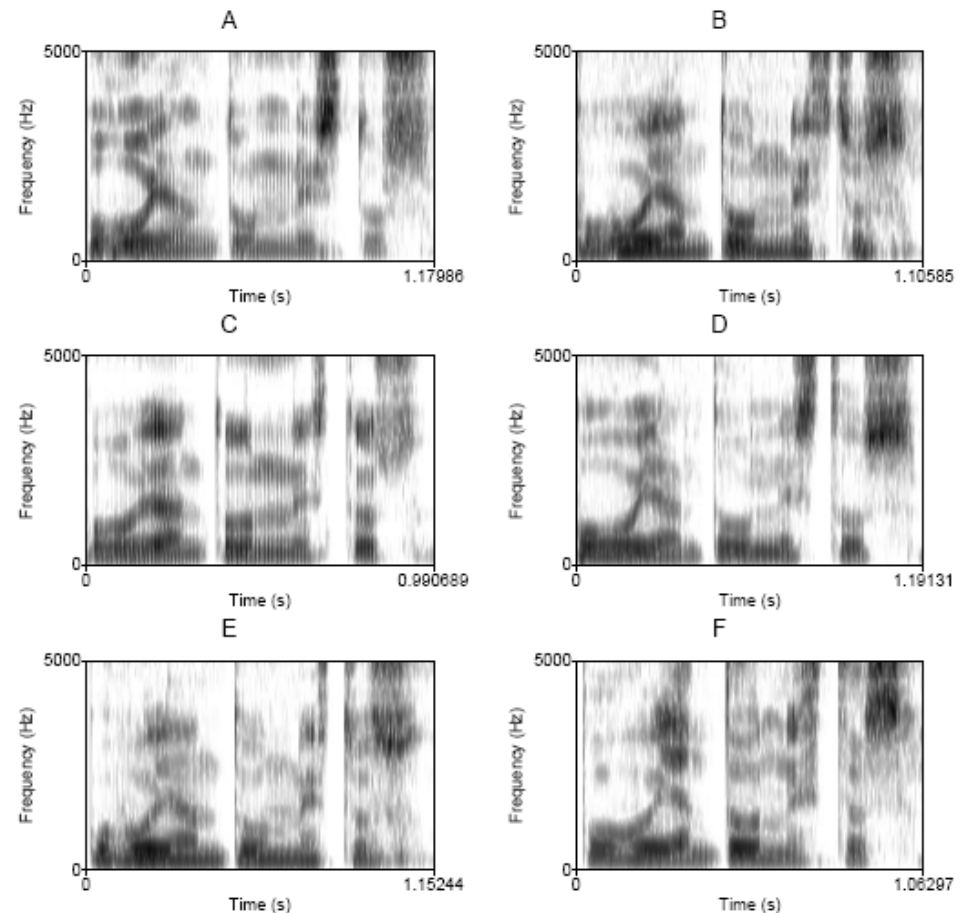
Example of pre-emphasis of a single frame [2]

# Feature Extraction

---

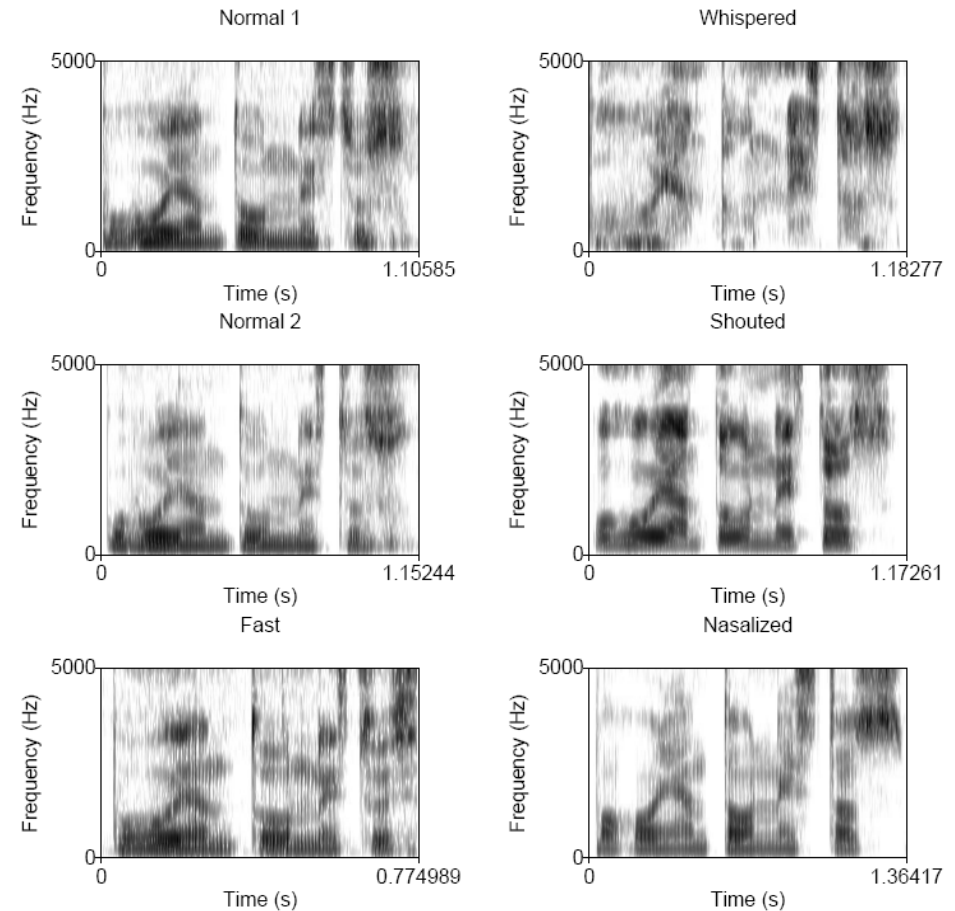
- Classification can be at most as accurate as the features extracted
- Transform the raw speech signal into compact, effective representation
- More stable and discriminative than the original signal

# Feature Extraction – contd.



Five different male speakers uttering the phrase “speaker recognition”. Two of the utterances are produced by the same speaker. [2]

# Feature Extraction – contd.



Six repetitions of the phrase  
 “speaker recognition” by the  
 same male speaker with  
 different voice styles [2]

# Feature Extraction – contd.

---

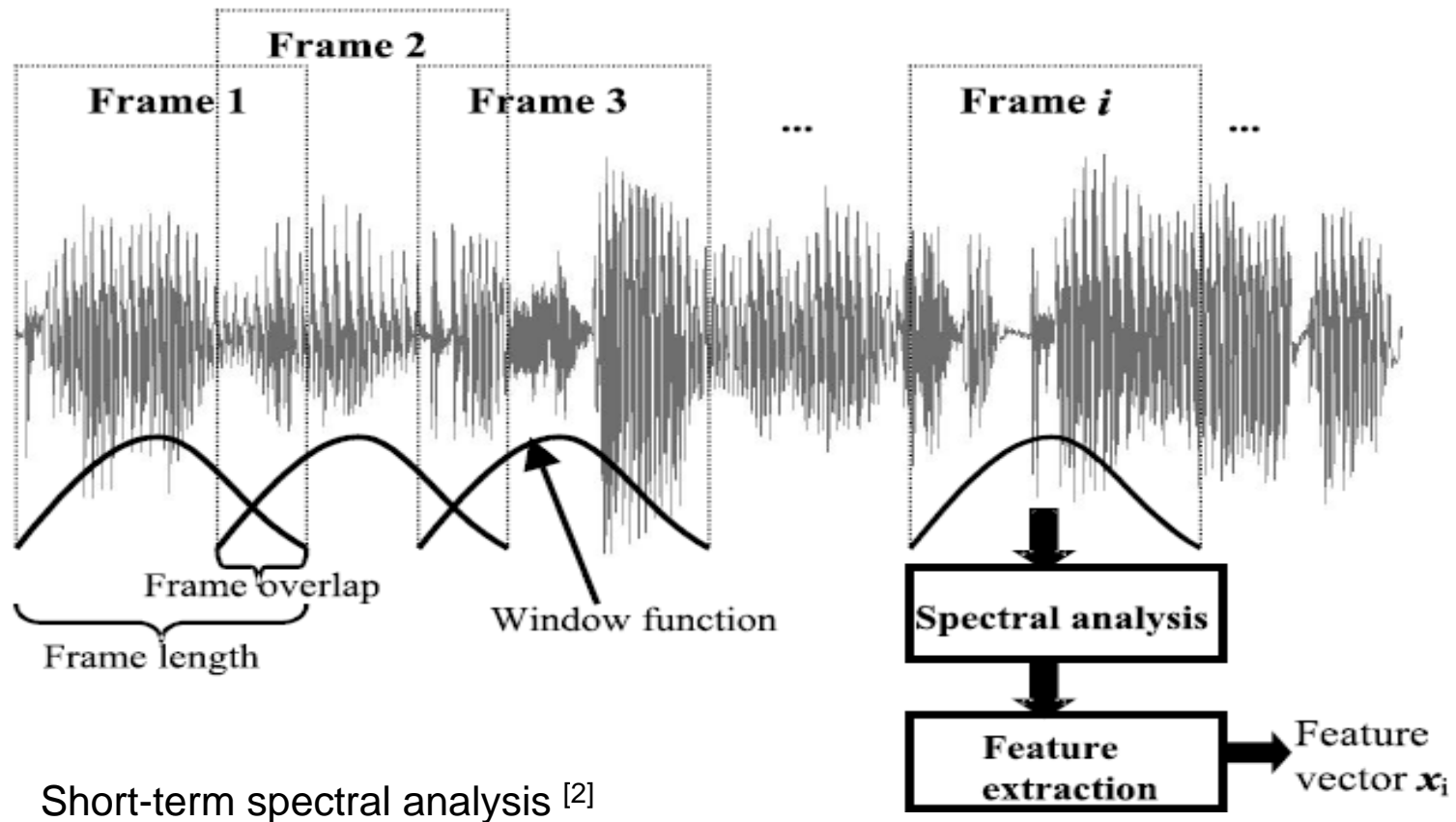
- High level features
  - Prosodics - syllable stress, rhyme, rhythm, intonation patterns
- Low level features
  - Frequency, cepstral coefficient based
  - Widespread use, easier to compute and model

# Feature Extraction – contd.

---

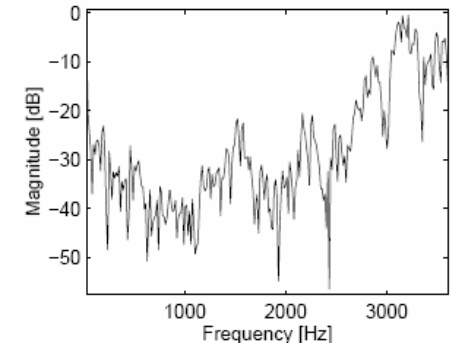
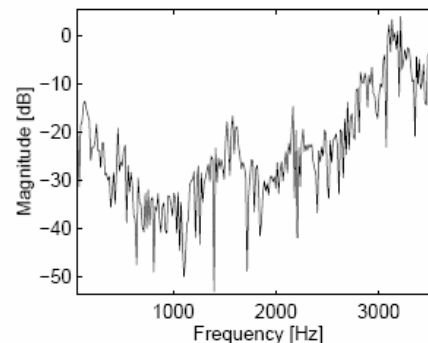
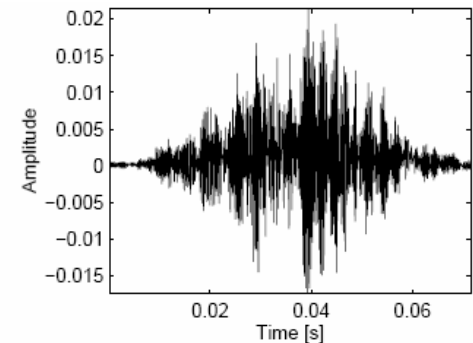
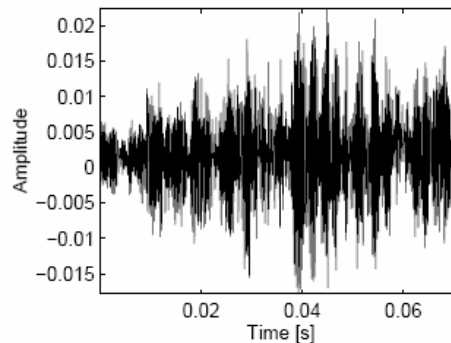
- **Short Term Spectral Analysis**
  - Speech signal changes continuously over time
  - Signal must be processed in short segments where parameters remain quasi-stationary
  - This is accomplished by *framing*
  - Frame length is crucial
  - Frame overlap is used to center each speech sound around some frame

# Feature Extraction – contd.



# Feature Extraction – contd.

- Short Term Spectral Analysis – contd.
  - Windowing is used to reduce the artifacts of framing
  - Rectangular window causes spectral leakage
  - Hamming window preserves higher-order harmonics



Speech segment windowed using Rectangular and Hamming windows [2]

# Feature Extraction – contd.

---

- **Cepstral Analysis**

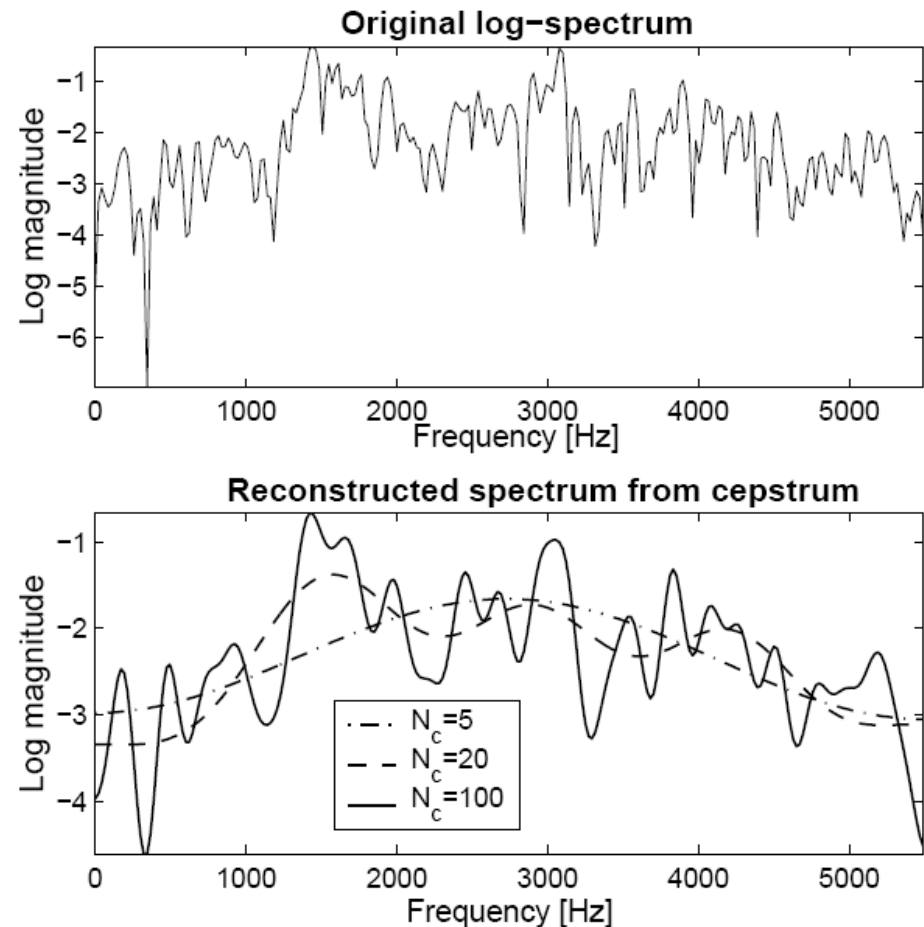
- Magnitude spectrum represented by combination of cosine basis functions of varying frequencies
- The cepstral co-efficients are the magnitudes of the basis functions
- The practical formula for computing cepstrum

$$c[n] = F^{-1} \{ \log |F \{ frame \}| \}$$

- *frame* is the windowed analysis frame
- Usually 12 co-efficients are used

# Feature Extraction – contd.

Example of spectrum reconstruction from cepstrum using different number of coefficients ( $N_c = 5; 20; 100$ ) [2]



# Codebook Generation

---

- Speaker model is called a codebook
- Vector Quantization method
  - The speaker models are formed by clustering the speaker's feature vectors in  $k$  non-overlapping clusters
  - Each cluster is represented by a *code vector*  $c_i$ , which is the centroid of the cluster
  - The resulting set of code vectors  $\{c_1, c_2, \dots, c_k\}$  is called a *codebook*

# Codebook Generation – contd.

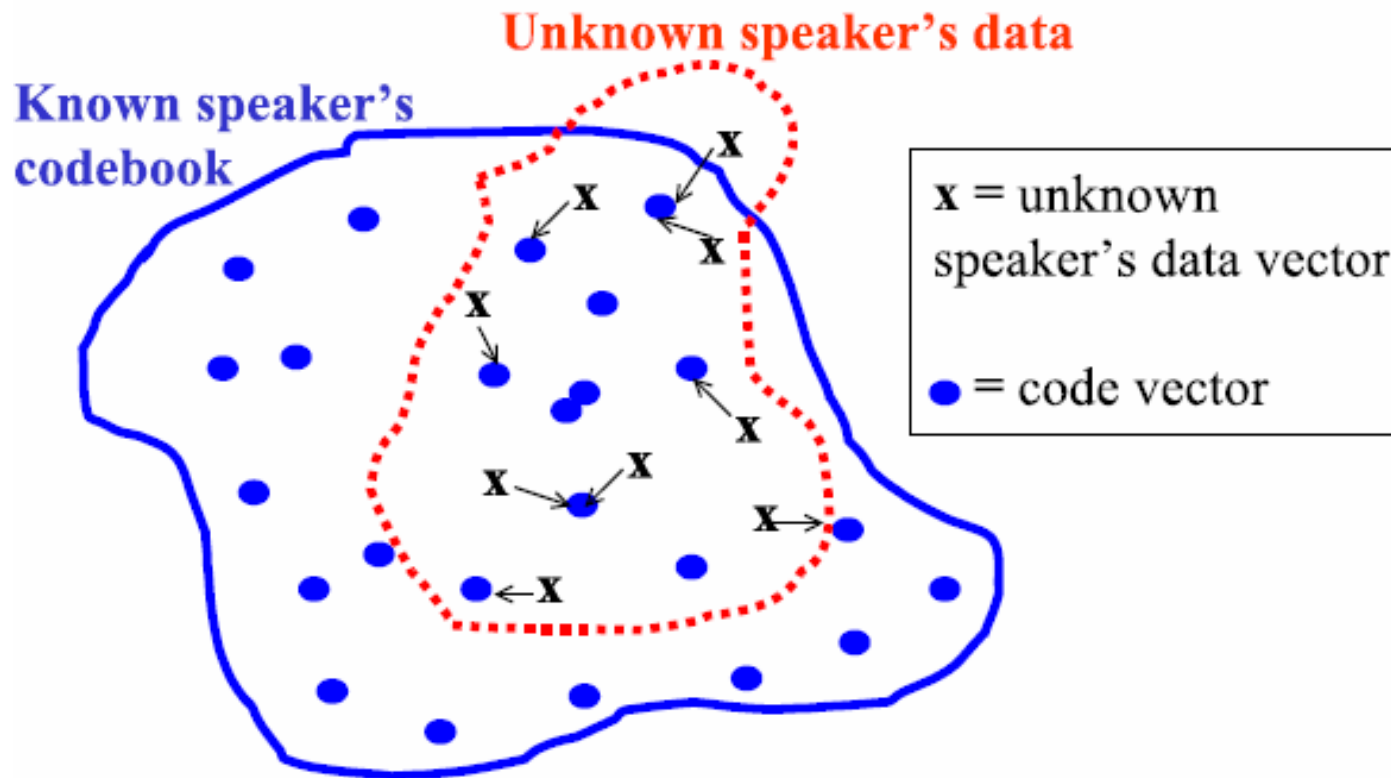
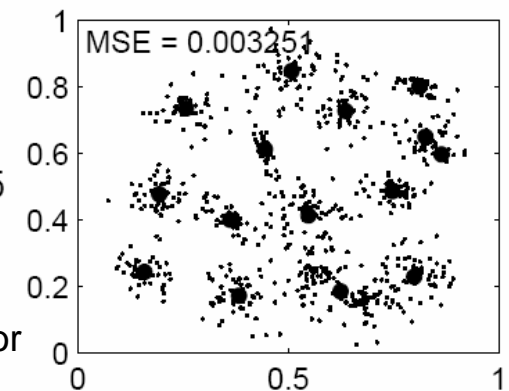
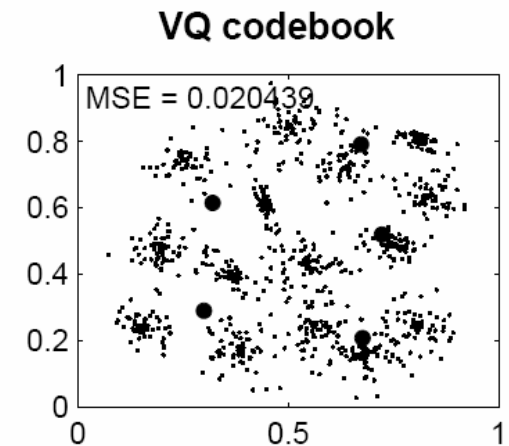


Illustration of VQ match score computation [2]

# Codebook Generation – contd.

- Linde, Buzo, Gray Algorithm
  - Input is the desired codebook size of  $K$  centers
  - LBG starts from initial codebook of size 1 (from  $k$ -means)  $K=5$
  - Refines codebook successively, splitting into twice as many clusters at each step
  - Cluster centers are given a random perturbation based on their MSE for that iteration  $K=15$



Examples of VQ based modeling for different model sizes ( $K = 5; 15$ ) [2]

# Codebook Generation – contd.

---

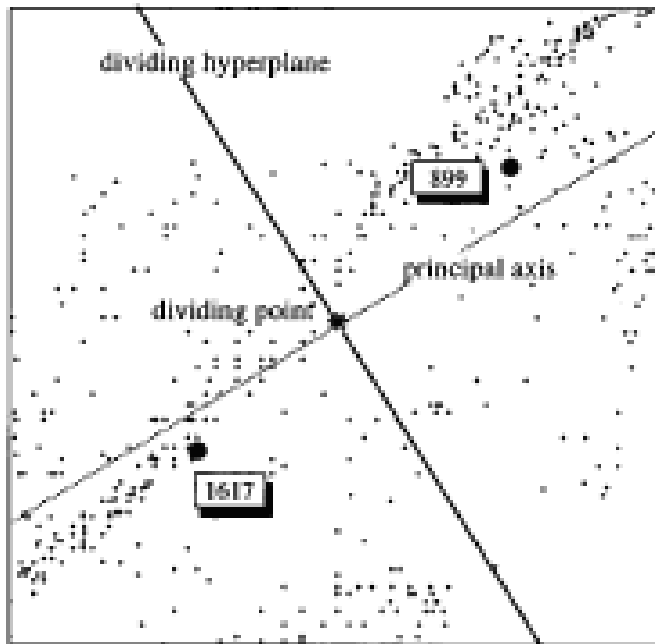
- **PCA Variant Algorithm**

Steps:

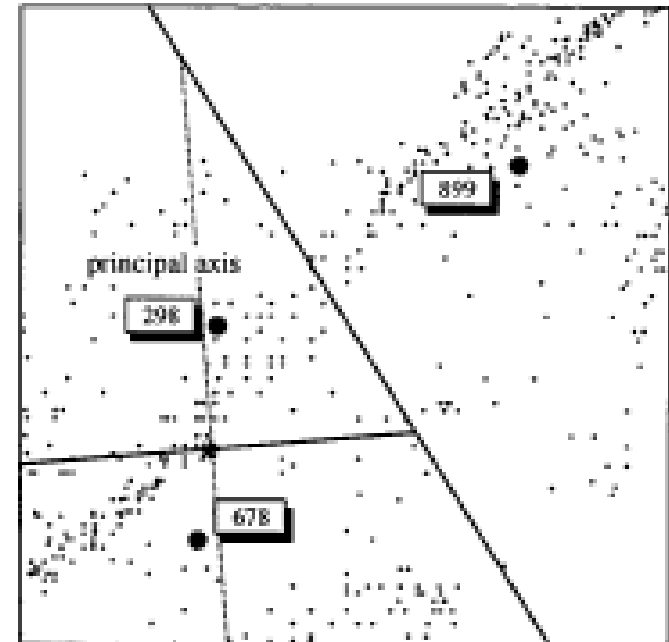
1. Calculate the principal axis using the power method.
2. Select the dividing point  $P$  on the principal axis.
3. Partition the training vectors with a hyperplane.
4. Calculate two new code vectors as the centroids of the two subclusters.

# Codebook Generation – contd.

- PCA Variant Algorithm – contd.



**Iteration = 1**



**Iteration = 2 [1]**

# Test Methodology

---

- Database Particulars
  - NIST 2001 Speaker Evaluation Database
  - 50 female speaker speech samples
  - 50 male speaker speech samples
  - Data format: Mono, 8KHz, Sphere format
- Training data generated by using first quarter of speech sample for each speaker and test data generated by using last quarter of speech sample for each speaker
- Training and test data are text independent

# Test Methodology – contd.

---

- Each training data speech sample is used to create a speaker codebook
- Codebook is characterized by method of creation and number of centroids
- The MSE of all feature vectors to the centroid it is closest to is computed and an overall MSE for all codebooks is generated.
- Each test data speech sample is pre-processed, features are extracted and is declared to be the speaker that closest matches that test sample

# Test Methodology – contd.

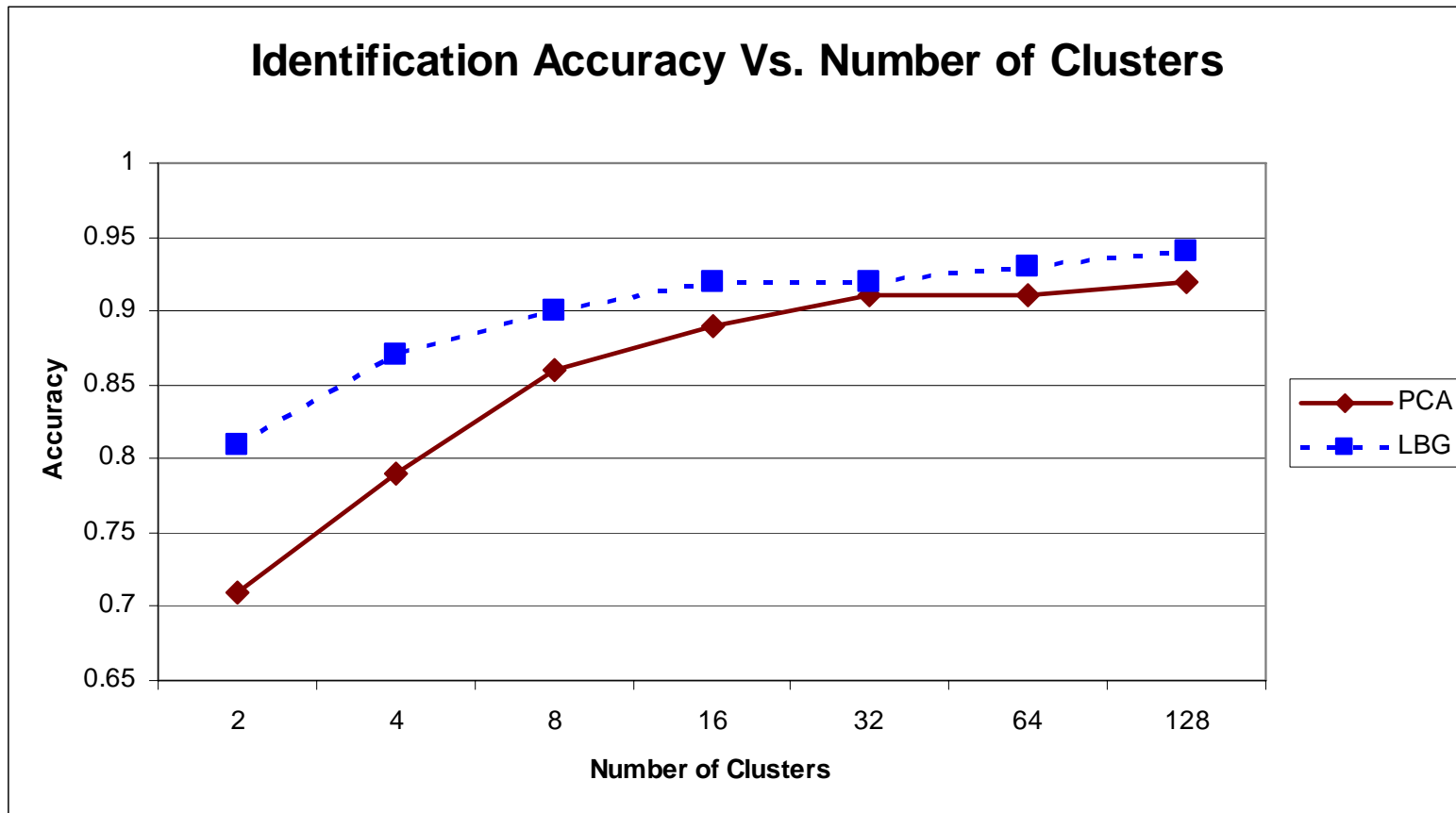
---

- Identification Procedure [1]
  - Compute feature vectors  $X = \{ \mathbf{x}_i \}$  for test speaker
  - **FOR EACH** speaker model  $C_i$  **DO**
    - Compute distortion,  $D_i = d(X, C_i)$  between  $X$  and  $C_i$
  - Identify the index of the unknown speaker  $Id$  as the one with the smallest distortion
    - $Id = \arg \min_{i=1toN} \{ D_i \}$
  - The distortion measure is defined as
    - $$d(X, C_i) = \frac{1}{L} \sum_{j=1}^L \min_{k=1}^K d_E(x_j, c_{ik})$$

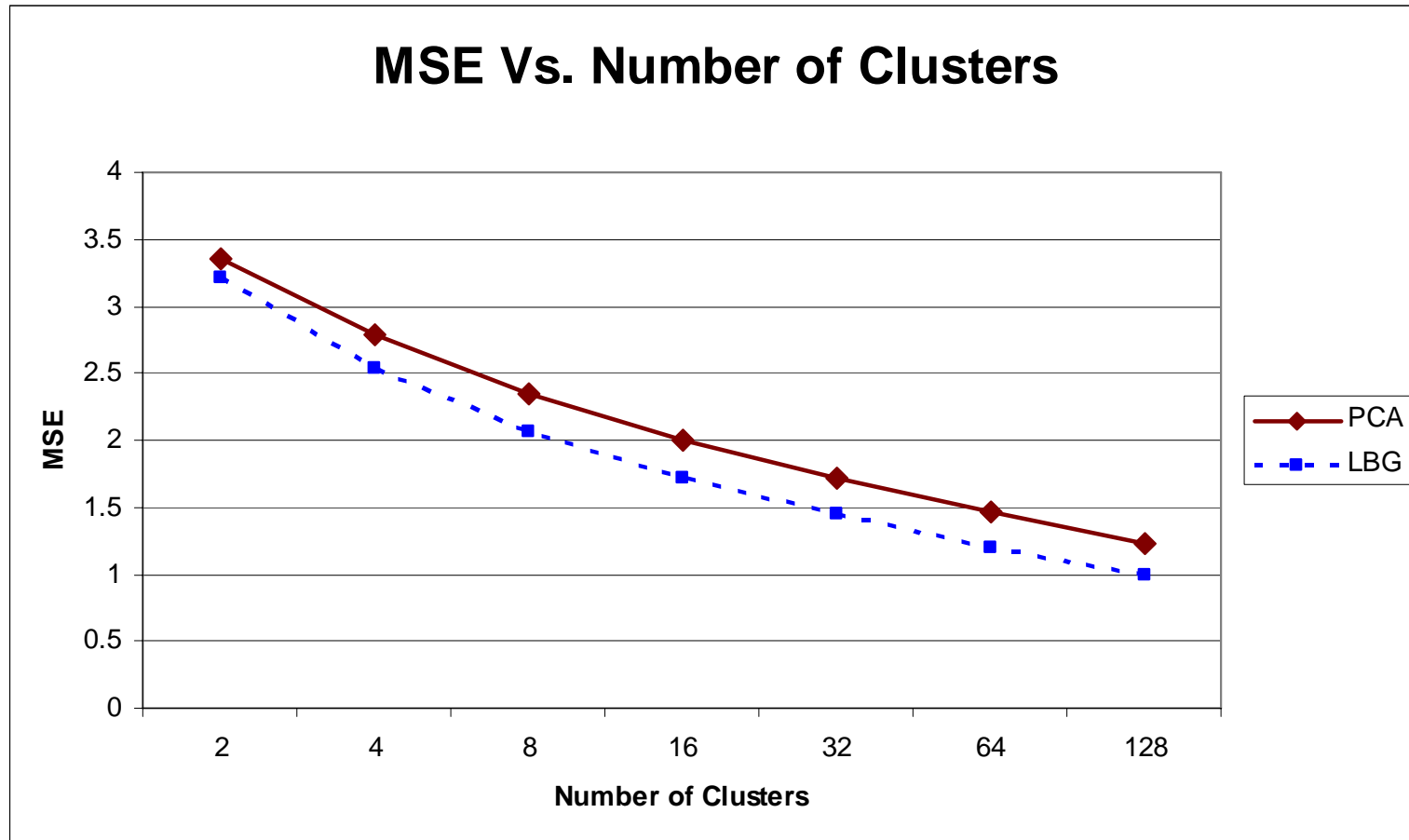
$$C_i = \{ c_{i1}, c_{i2}, \dots, c_{iK} \}$$

$$X = \{ x_1, x_2, \dots, x_L \}$$

# Experimental Results

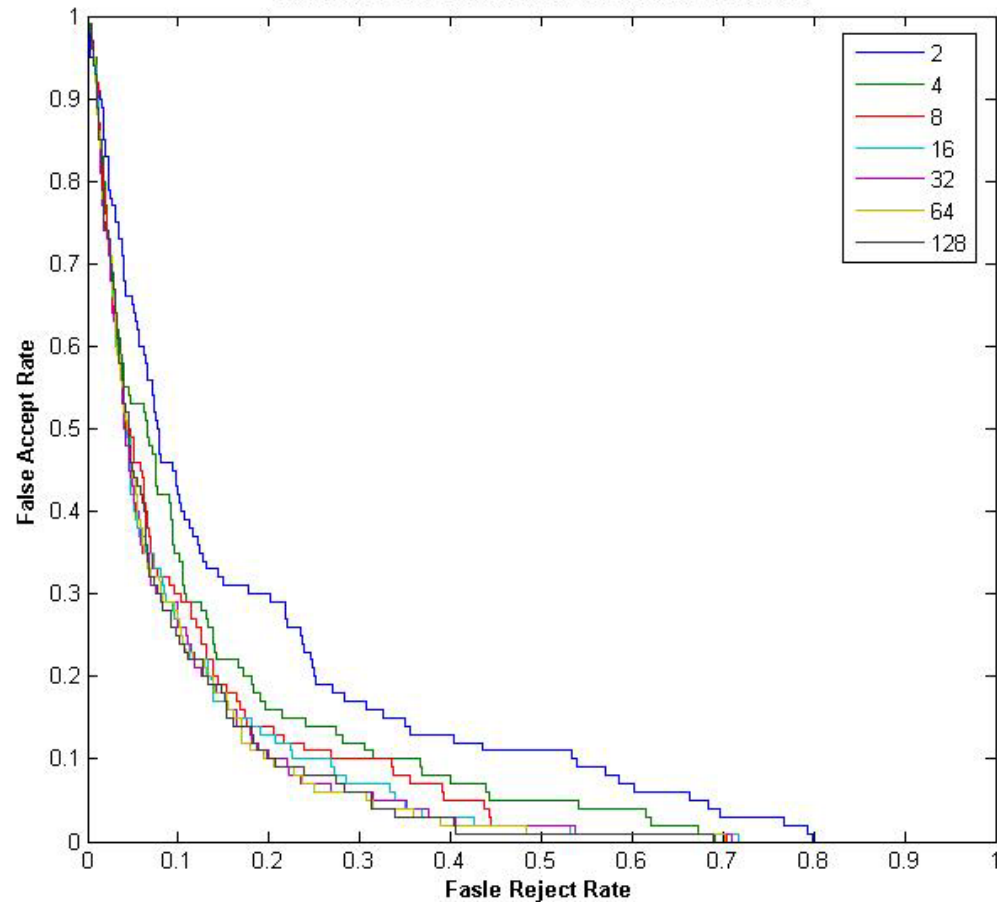


# Experimental Results – contd.



# Experimental Results – contd.

ROC for LBG Method for different Clusters



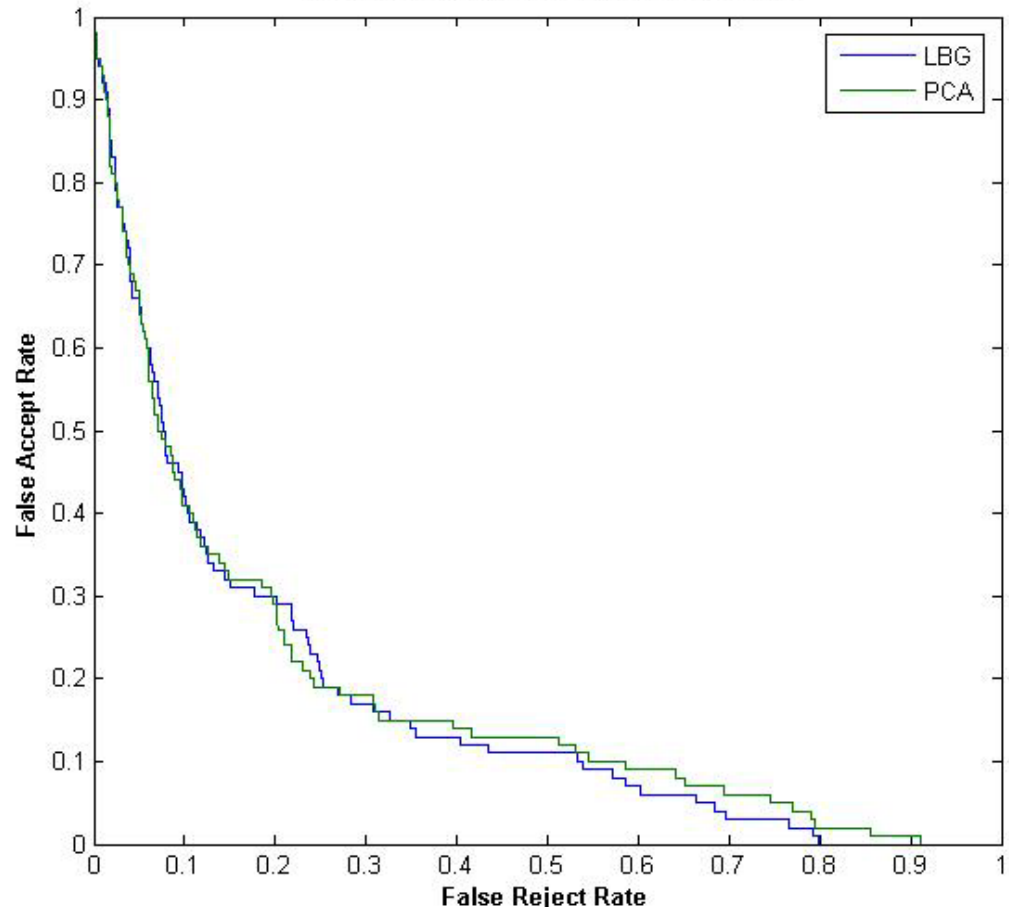
ROC for LBG method

Score =  $1 / d(X, C_i)$

Min-max Normalization

# Experimental Results – contd.

ROC for LBG and PCA at 2 Clusters



ROC for LBG and PCA  
method at 2 Clusters

Score =  $1 / d(X, C_i)$

Min-max Normalization

# References

---

- [1] Tomi Kinnunen, Teemu Kilpelainen And Pasi Franti, “**Comparison Of Clustering Algorithms In Speaker Identification**”
- [2] Tomi Kinnunen, “**Spectral Features for Automatic Text-Independent Speaker Recognition**”
- [3] Fränti P., Kaukoranta T., Nevalainen O., “**On The Splitting Method For Vector Quantization Codebook Generation**”, *Optical Engineering*, 36(11): pp. 3043-3051, November 1997
- [4] Furui S., “**Cepstral Analysis Technique for Automatic Speaker Verification**”; *IEEE Trans. on Acoustics, Speech and Signal Processing*, 29(2) pp. 254-272, 1981
- [5] Linde Y., Buzo A., Gray R.M., “**An Algorithm For Vector Quantizer Design**”; *IEEE Trans. On Communications*, 28(1) pp. 84-95, January 1980

# Questions ?

---