ISSN 2249-6343
International Journal of Computer Technology and Electronics Engineering (IJCTEE)
Volume 3, Special Issue, March-April 2013, An ISO 9001: 2008 Certified Journal.
E-NSPIRE, A National Level Conference held at Pravara Rural Engineering College, Loni, Maharashtra, INDIA.

Vector Quantization Approach for Speaker Recognition

Hemlata Eknath Kamale, Dr.R. S. Kawitkar

Abstract: Speaker recognition is the process of automatically recognizing who is speaking on the basis of individual information included in speech waves. This technique makes it possible to use the speaker's voice to verify their identity and control access to services such as voice dialing, banking by telephone, telephone shopping, database access services, information services, voice mail, security control for confidential information areas, and remote access to computers. Speaker-specific characteristics exist in speech signals due to different speakers having different resonances of the vocal tract. These differences can be exploited by extracting feature vectors such as Mel-Frequency Cepstral Coefficients (MFCCs) from the speech signal. The Vector Quantization (VQ) approach is used for mapping vectors from a large vector space to a finite number of regions in that space. Each region is called a cluster and can be represented by its center called a codeword. The collection of all code words is called a codebook. After the enrollment session, the acoustic vectors extracted from input speech of a speaker provide a set of training vectors. LBG algorithm due to Linde, Buzo and Gray is used for clustering a set of L training vectors into a set of M codebook vectors.

***Index terms*- Automatic speaker recognition (ASR), Mel frequency cepstral coefficient (MFCC), Speech processing, Speaker verification.**

I. INTRODUCTION

Speaker recognition is an important branch of speech processing. It is the process of automatically recognizing who is speaking by using speaker-specific information included in the speech waveform. It has two phases:- Enrollment session or Training phase: In the training phase, each registered speaker has to provide samples of their speech so that the system can build or train a reference model for that speaker. Operation session or testing phase: During the testing (operational) phase, the input speech is matched with stored reference model and recognition decision is made [2]. Speaker recognition is a process that enables machines to understand and interpret the human speech by making use of certain algorithms and verifies the authenticity of a speaker with the help of a database. That is, speaker recognition or identification is essentially a method of automatically identifying a speaker from a recorded or a live speech signal by analyzing the speech signal parameters. First, the human speech is converted to machine readable format after which the machine processes the data. The data processing deals with feature extraction and feature matching. Then, based on the processed data, suitable action is taken by the machine. The action taken depends on the application. Every speaker is identified with the help of unique numerical values of certain signal parameters called 'template' or 'code book' pertaining to the speech produced by his or her vocal tract. Normally the speech parameters of a vocal tract that are considered for analysis are (i) formant frequencies, (ii) pitch, and (iii) loudness [3].

II. LITERATURE SURVEY

Each speaker recognition system has two phases: Enrollment and verification. During enrollment, the speaker's voice is recorded and typically a number of features are extracted to form a voice print, template, or model. In the verification phase, a speech sample or "utterance" is compared against a previously created voice print. For identification systems, the utterance is compared against multiple voice prints in order to determine the best match(es) while verification systems compare an utterance against a single voice print. Because of the process involved, verification is faster than identification. Speaker recognition systems fall into two categories: text-dependent and text-independent. Text-Dependent: If the text must be the same for enrollment and verification this is called text-dependent recognition. In a text-dependent system, prompts can either be common across all speakers (e.g.: a common pass phrase) or unique. In addition, the use of shared-secrets (e.g. passwords and PINs) or knowledge-based information can be employed in order to create a multi-factor authentication scenario. Text-Independent: Text-independent systems are most often used for speaker identification as they require very little if any cooperation by the speaker. In this case the text during enrollment and test is different. In fact, the enrollment may happen without the user's knowledge, as in the case for many forensic applications. As text-independent technologies do not compare what was said at enrollment and verification, verification applications tend to also employ speech recognition to determine what the user is saying at the point of authentication. In text independent systems both acoustics and speech analysis techniques are used. There are different techniques for feature extraction, the most common being linear predictive coding (LPC) and Mel frequency cepstral coefficients (MFCC). LPC is a time-domain technique and suffers from variations in the amplitude of the speech signal due to noise. The preferred technique for feature extraction is MFCC wherein the features are generated by transforming the signal into frequency domain [1]. In general, cepstral features are more compact, discriminable, and most importantly, nearly decorrelated and therefore, they can provide higher baseline performance over filter bank features. There are some of the popular models in text-independent speaker recognition as follows.

ISSN 2249-6343
International Journal of Computer Technology and Electronics Engineering (IJCTEE)
Volume 3, Special Issue, March-April 2013, An ISO 9001: 2008 Certified Journal.
E-NSPIRE, A National Level Conference held at Pravara Rural Engineering College, Loni, Maharashtra, INDIA.

A. Vector Quantization

Vector quantization (VQ) model also known as centroid model, is one of the simplest text-independent speaker models. It was introduced to speaker recognition in the 1980s and its roots are originally in data compression. Even though VQ is often used for computational speedup techniques and lightweight practical implementations, it also provides competitive accuracy when combined with background model adaptation [4].

B. Gaussian Mixture Model

Gaussian mixture model (GMM) is a stochastic model which has become the de facto reference method in speaker recognition. The GMM can be considered as an extension of the VQ model, in which the clusters are overlapping. A GMM is composed of a finite mixture of multivariate Gaussian components.

C. Support Vector Machine

Support vector machine (SVM) is a powerful discriminative classifier that has been recently adopted in speaker recognition. It has been applied both with spectral, prosodic, and high-level features. Currently SVM is one of the most robust classifiers in speaker verification, and it has also been successfully combined with GMM to increase accuracy. One reason for the popularity of SVM is its good generalization performance to classify unseen data.

D. Artificial neural networks

ANNs have been used in various pattern classification problems, including speaker recognition. A potential advantage of ANNs is that feature extraction and speaker modeling can be combined into a single network, enabling joint optimization of the (speaker-dependent) feature extractor and the speaker model.

III. SYSTEM OVERVIEW

Speaker recognition can be classified into identification and verification. Speaker identification is the process of determining which registered speaker provides a given utterance. Speaker verification, on the other hand, is the process of accepting or rejecting the identity claim of a speaker. Figure 1 shows the basic structures of speaker identification and verification systems. The system that we will describe is classified as text-independent speaker identification system since its task is to identify the person who speaks regardless of what is saying. At the highest level, all speaker recognition systems contain two main modules feature extraction and feature matching. Feature extraction is the process that extracts a small amount of data from the voice signal that can later be used to represent each speaker. Feature matching involves the actual procedure to identify the unknown speaker by comparing extracted features from his/her voice input with the ones from a set of known speakers. We will discuss each module in detail in later sections.

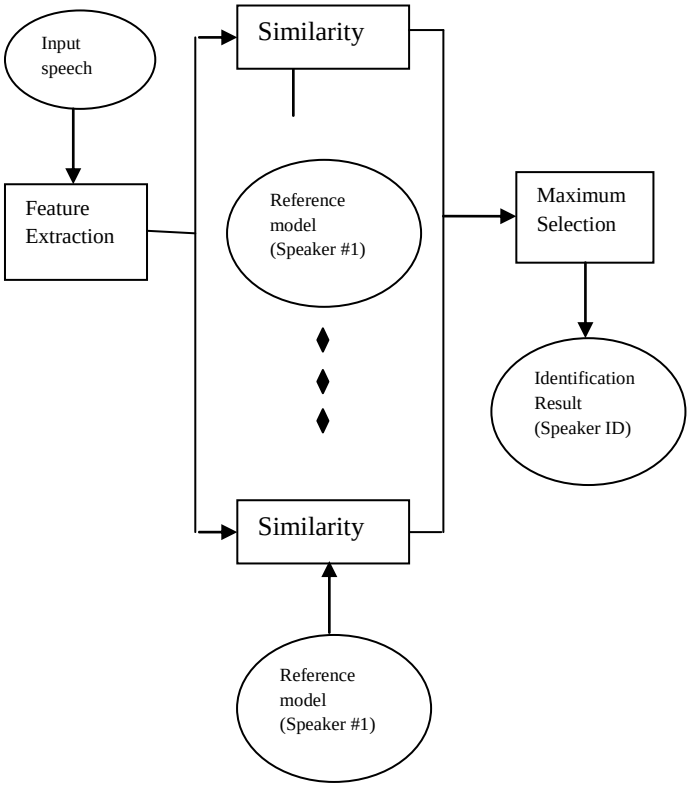

Fig. A) Speaker Identification

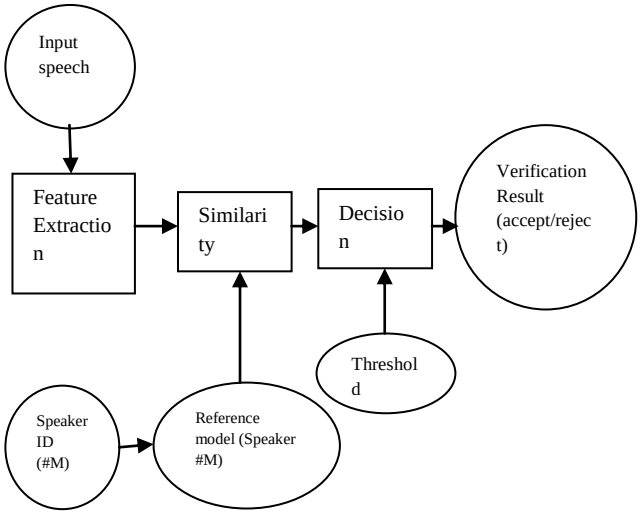

Fig. b) Speaker Verification

Fig. 1 Basic Structures of Speaker Recognition Systems

A. Feature Extraction

The purpose of feature extraction is to convert the speech waveform to a set of features for further analysis. This is often referred to as the signal processing front end. The speech signal is a slowly time-varying signal and when it is examined over a sufficiently short period of time, its characteristics are fairly stationary, but over long periods of time the signal characteristics change to reflect the different speech sounds being spoken. In many cases, short-time spectral analysis is the most common way to characterize the speech signal.

ISSN 2249-6343
International Journal of Computer Technology and Electronics Engineering (IJCTEE)
Volume 3, Special Issue, March-April 2013, An ISO 9001: 2008 Certified Journal.
E-NSPIRE, A National Level Conference held at Pravara Rural Engineering College, Loni, Maharashtra, INDIA.

Several possibilities exist for parametrically representing the speech signal for the speaker identification task, such as MFCCs, Linear Prediction Coding (LPC), and others [5]. In this system, MFCCs are chosen because they are based on the perceptual characteristics of the human auditory system. The process of computing MFCCs is described in more detail next.

1 The Concept of MFCC

The Mel-Frequency Cepstrum (MFC) is a representation of short-term power spectrum of a sound. The MFCCs are coefficients that collectively make up an MFC. They are derived from a type of cepstral representation of the audio clip (a nonlinear "spectrum-of-a-spectrum"). The difference between the cepstrum and the mel frequency cepstrum is that in the MFC, the frequency bands are equally spaced on the mel scale, which approximates the human auditory system's response more closely than the linearly-spaced frequency bands used in the normal cepstrum. The cepstrum is a common transform used to gain information from a person's speech signal. It can be used to separate the excitation signal (which contains the words and the pitch) and the transfer function (which contains the voice quality). It is the result of taking Fourier transform of decibel spectrum as if it were a signal. We use cepstral analysis in speaker identification because the speech signal is of the particular form above, and the "cepstral transform" of it makes analysis simple. Mathematically, cepstrum of signal = FT[log{FT(the windowed signal)}] The cepstrum can be seen as information about rate of change in the different spectrum bands. It is now used as an excellent feature vector for representing the human voice and musical signals.

2 Calculation of MFCC

MFCCs are commonly calculated by first taking the Fourier transform of a windowed excerpt of a signal and mapping the powers of the spectrum obtained above onto the mel scale, using triangular overlapping windows. Next the logs of the powers at each of the mel frequencies are taken, Direct Cosine Transform is applied to it (as if it were a signal). The MFCCs are the amplitudes of the resulting spectrum. This procedure is represented step-wise in the figure below.

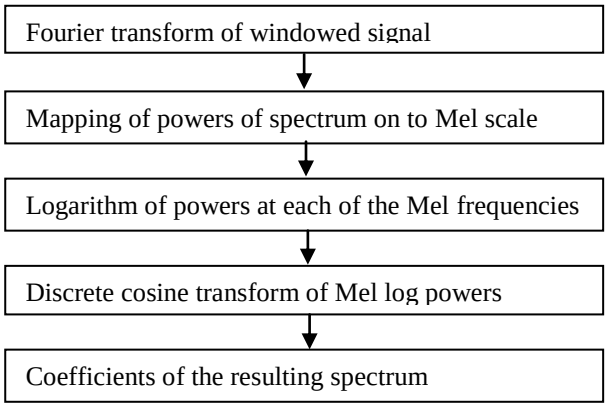

Fig. 2. Flow Diagram to Calculate MFCC [3]

1) *Feature matching*

The problem of speaker recognition belongs to a much broader topic in scientific and engineering so called pattern recognition. The goal of pattern recognition is to classify objects of interest into one of a number of categories or classes. The objects of interest are generically called patterns and in our case are sequences of acoustic vectors that are extracted from an input speech. The classes here refer to individual speakers. Since the classification procedure in our case is applied on extracted features, it can be also referred to as feature matching. Furthermore, if there exists some set of patterns that the individual classes of which are already known, then one has a problem in supervised pattern recognition. This is exactly our case since during the training session, we label each input speech with the ID of the speaker. These patterns comprise the training set and are used to derive a classification algorithm. The remaining patterns are then used to test the classification algorithm; these patterns are collectively referred to as the test set. If the correct classes of the individual patterns in the test set are also known, then one can evaluate the performance of the algorithm. The state-of-the-art in feature matching techniques used in speaker recognition includes Dynamic Time Warping (DTW), Hidden Markov Modeling (HMM), and Vector Quantization (VQ). In this paper, the VQ approach is used, due to ease of implementation and high accuracy. VQ is a process of mapping vectors from a large vector space to a finite number of regions in that space. Each region is called a cluster and can be represented by its center called a codeword. The collection of all code words is called a codebook.

2 Vector Quantization

The advantages of VQ are:

- Reduced storage for spectral analysis information.
- Reduced computation for determining similarity of spectral analysis vectors. In speech recognition, a major component of the computation is the determination of spectral similarity between a pair of vectors. Based on the VQ representation this is often reduced to a table lookup of similarities between pairs of codebook vectors.
- Discrete representation of speech sounds [2].

Below figure shows the block diagram of a speaker recognition model using VQ

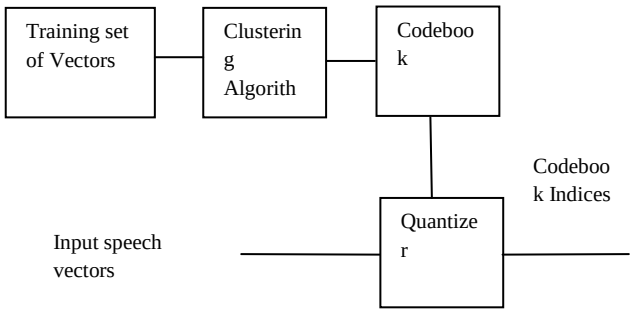

Fig.3. Block diagram of basic VQ training and classification structure [2]

ISSN 2249-6343
International Journal of Computer Technology and Electronics Engineering (IJCTEE)
Volume 3, Special Issue, March-April 2013, An ISO 9001: 2008 Certified Journal.
E-NSPIRE, A National Level Conference held at Pravara Rural Engineering College, Loni, Maharashtra, INDIA.

Vector quantization (VQ in short) involves the process of taking a large set of feature vectors of a particular user and producing a smaller set of feature vectors that represent the centroids of the distribution, i.e. points spaced so as to minimize the average distance to every other point. Vector quantization is used since it would be highly impractical to represent every single feature vector in feature space that we generate from the training utterance of the corresponding speaker. While the VQ algorithm does take a while to generate the centroids, it saves a lot of time during the testing phase as we are only considering few feature vectors instead of overloaded feature space of a particular user. Therefore is an economical compromise that we can live with. A vector quantizer maps k-dimensional vectors in the vector space *Rk* into a finite set of vectors Y = {yi: *i* = 1, 2, ..., *N*}. Here k-dimension refers to the no of feature coefficients in each feature vector. Each vector y*i* is called a code vector or a codeword and the set of all the code words is called a codebook [2].

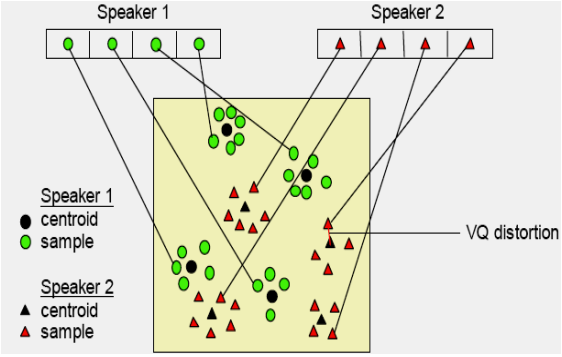

Fig 4. Vector Quantization Codebook Formation [3]

Above figure shows a conceptual diagram to illustrate this recognition process. In the figure, only two speakers and two dimensions of the acoustic space are shown. The circles refer to the acoustic vectors from the speaker 1 while the triangles are from the speaker 2. In the training phase, a speaker-specific VQ codebook is generated for each known speaker by clustering his/her training acoustic vectors. The result code words (centroids) are shown in Figure by black circles and black triangles for speaker 1 and 2, respectively. The distance from a vector to the closest codeword of a codebook is called a VQ-distortion. In the recognition phase, an input utterance of an unknown voice is "vector-quantized" using each trained codebook and the total VQ distortion is computed. The speaker corresponding to the VQ codebook with smallest total distortion is identified.

3) Optimization using LBG algorithm

After the enrollment session, the feature vectors extracted from input speech of each speaker provide a set of training vectors for that speaker. The next important task is to build a speaker-specific VQ codebook for each speaker using the training vectors extracted [2]. There is a well-known algorithm, namely LBG algorithm [Linde, Buzo and Gray, 1980], for clustering a set of L training vectors into a set of M codebook vectors. The algorithm is formally implemented by the following procedure:-

1. Design a 1-vector codebook; this is the centroid of the entire set of training vectors (hence, no iteration is required here).
2. Increase the size of the codebook twice by splitting each current codebook y_n according to the rule

$$y_n^+ = y_n(1+\varepsilon)$$

$$y_n^- = y_n(1-\varepsilon)$$

where n varies from 1 to the current size of the codebook, and ε is a splitting parameter (we choose ε =0.01).
3. Nearest-Neighbor Search: for each training vector, find the codeword in the current codebook that is the closest (in terms of similarity measurement), and assign that vector to the corresponding cell (associated with the closest codeword).
4. Centroid Update: update the codeword in each cell using the centroid of the training vectors assigned to that cell.
5. Iteration 1: repeat steps 3 and 4 until the average distance falls below a preset threshold
6. Iteration 2: repeat steps 2, 3 and 4 until a codebook size of *M* is designed. Intuitively, the LBG algorithm generates an *M*-vector codebook iteratively. It starts first by producing a 1-vector codebook, then uses a splitting technique on the codeword to initialize the search for a 2-vector codebook, and continues the splitting process until the desired *M*-vector codebook is obtained [2].

V. CONCLUSION

The system is based on identifying an unknown speaker from given a set of registered speakers. Here we have assumed the unknown speaker to be one of the known speakers and tried to develop a model to which it can best fit into. In the first step of generating the speaker recognition model, we went for feature extraction using Mel Frequency Cepstral Coefficients These features act as a basis for further development of the speaker identification process. Next we went for feature mapping using the vector Quantization using LBG algorithm. The results obtained using MFCC and VQ are appreciable. MFCCs for each speaker were computed and vector quantized for efficient representation. The code books were generated using LBG algorithm which optimizes the quantization process. VQ distortion between the resultant codebook and MFCCs of an unknown speaker was taken as the basis for determining the speaker's authenticity. Accuracy of 75% was obtained using VQLBG algorithm.

REFERENCES

[1] Santosh V. Chapaneri "Spoken Digits Recognition using Weighted MFCC and Improved Features for Dynamic Time Warping" International Journal of Computer Applications (0975 – 8887) Volume 40– No.3, February 2012.

ISSN 2249-6343
International Journal of Computer Technology and Electronics Engineering (IJCTEE)
Volume 3, Special Issue, March-April 2013, An ISO 9001: 2008 Certified Journal.
E-NSPIRE, A National Level Conference held at Pravara Rural Engineering College, Loni, Maharashtra, INDIA.

[2] Ashish Kumar Panda, Amit Kumar Sahoo "Study of speaker recognition systems" Department Of Electronics And Communication National Institute Of Technology, Rourkela 2007.

[3] Prof. Ch.Srinivasa Kumar, Dr. P. Mallikarjuna Rao "Design of an automatic speaker recognition system using MFCC, vector quantization and LBG algorithm" Ch.Srinivasa Kumar et al./ International Journal on Computer Science and Engineering (IJCSE) vol. 3 No. 8 August 2011.

[4] Tomi Kinnunen_,a, Haizhou Lib "An overview of text-independent speaker recognition: from features to Super vectors" Speech Communication July 1, 2009.

[5] Phaklen EhKan,1, 2 Timothy Allen,1 and Steven F. Quigley1 " FPGA implementation for GMM-based speaker identification" Hindawi Publishing Corporation International Journal of Reconfigurable Computing volume 2011, Article ID 420369, 8 pages doi:10.1155/2011/420369.

[6] Joseph P. Campbell, Jr. Department of Defense Fort Meade, "MD speaker recognition".

[7] Martin Cooke, Phil Green and Malcolm Crawford "Handling missing data in speech recognition" presented at the International Conference On Spoken Language Processing (Icslp-94), Yokahama, Japan.

[8] D. Reynolds and R. Rose, "Robust text-independent speaker identification using Gaussian mixture speaker models," IEEE Trans. Speech Audio Process., vol. 3, no. 1, pp. 72–83, Jan. 1995.

[9] M. Cooke, P. Green, L. Josifovski, and A. Vizinho, "Robust automatic speech recognition with missing and unreliable acoustic data," Speech Commun., vol. 34, pp. 267–285, 2001.

[10] D. A. Reynolds, T. F. Quatieri, and R. B. Dunn, "Speaker verification using adapted Gaussian mixture models," Digital Signal Process., vol. 10, pp. 19–41, 2000.

[11] R. P. Lippmann, "Speech recognition by machines and humans," Speech Commun., vol. 22, pp. 1–15, 1997.

[12] B. S. Atal, "Effectiveness of linear prediction characteristics of the speech wave for automatic speaker identification and verification," J. Acoust. Soc. Amer., vol. 55, no. 6, pp. 1304–1312, 1974.

[13] H. Hermansky and N. Morgan, "RASTA processing of speech," IEEE Trans. Speech Audio Process, vol. 2, no. 4, pp. 578–589, Oct. 1994.

[14] Raj B. and Stern R."Missing-feature approaches in speech recognition", IEEE Signal Proc. Magazine, 2005.

[15] An Automatic Speaker Recognition System http://www.ifp.uiuc.edu/~minhdo/teaching/speaker_recognition.