MFCC Feature Extraction

Speech Recognition

Voice or speaker recognition is the ability of a machine or program to receive and interpret dictation or to understand and carry out spoken commands.

Voice or speaker recognition software on computers requires that analog audio be converted into digital signals, known as analog-to-digital conversion. For a computer to decipher a signal, it must have a digital database, or vocabulary, of words or syllables, as well as a speedy means for comparing this data to signals. The speech patterns are stored on the hard drive and loaded into memory when the program is run. A comparator checks these stored patterns against the output of the A/D converter this is an action called pattern recognition.

Some speech recognition systems require "training" (also called "enrollment") where an individual speaker reads text or isolated vocabulary into the system. The system analyzes the person's specific voice and uses it to fine-tune the recognition of that person's speech, resulting in increased accuracy. Systems that do not use training are called "speaker independent" systems. Systems that use training are called "speaker dependent" systems.

Speech Recognition History

1950s and 1960s: Baby Talk;
The first speech recognition systems could understand only digits. (Given the complexity of human language, it makes sense that inventors and engineers first focused on numbers.)Bell Laboratories designed in 1952 the "Audrey" system, which recognized digits spoken by a single voice. Ten years later, IBM demonstrated at the 1962 World's Fair its "Shoebox"machine, which could understand 16 words spoken in English.

Also around this time Soviet researchers invented the dynamic time warping (DTW) algorithm and used it to create a recognizer capable of operating on a 200-word vocabulary. The DTW algorithm processed the speech signal by dividing it into short frames, e.g. 10ms segments, and processing each frame as a single unit. Although DTW would be superseded by later algorithms, the technique of dividing the signal into frames would carry on. Achieving speaker independence was a major unsolved goal of researchers during this time period.

1970s: Speech Recognition Takes Off;
Speech recognition technology made major strides in the 1970s, thanks to interest and funding from the U.S. Department of Defense. The DoD's DARPA Speech Understanding Research (SUR) program, from 1971 to 1976, was one of the largest of its kind in the history of speech recognition, and among other things it was responsible for Carnegie Mellon's

“Harpy” speech-understanding system. Harpy could understand 1000 words, approximately the vocabulary of an average three-year-old child.

1980s:
Speech Recognition Turns Toward Prediction Over the next decade, thanks to new approaches to understanding what people say, speech recognition vocabulary jumped from about a few hundred words to several thousand words, and had the potential to recognize an unlimited number of words. One major reason was a new statistical method known as the Hidden Markov Model.

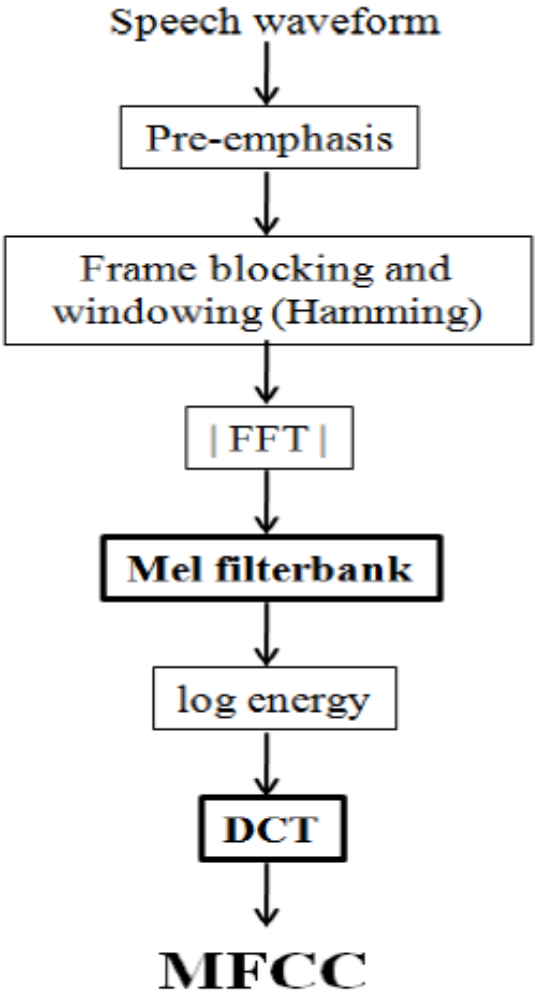
Speech Recognition Process

Feature extraction process

- Pre-emphasis
- Framing
- Windowing
- FFT
- Mel Cepstrum
- Cepstral Lifting

Feature Matching

- Hidden Markow Model (HMM)



MFCC extraction process

Feature extraction

Feature Analysis Approach

Feature Analysis Approach systems are usually speaker independent. Instead of finding an exact or a near match to the actual voice template and the stored template, this method first processes the voice using the Fast Fourier Transforms or the Linear Predictive Coding. In the next step the

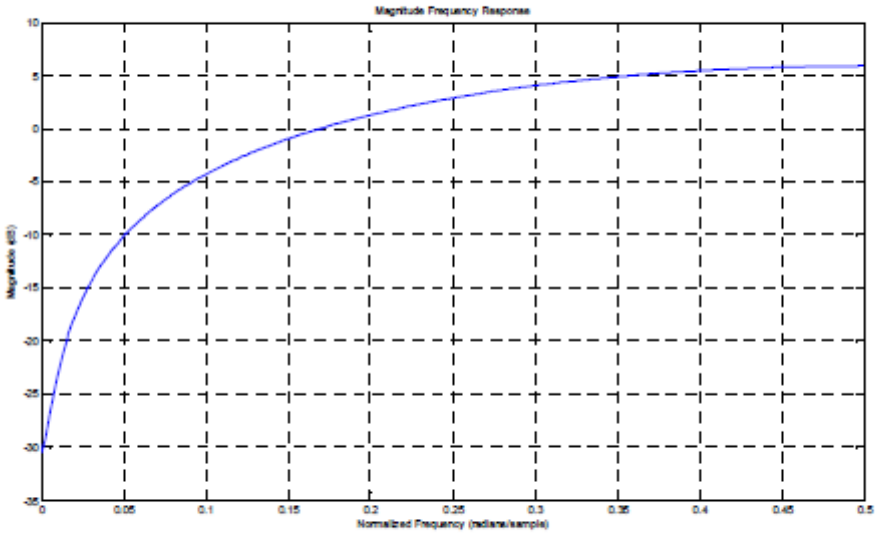
system tries to find the similarities between the expected inputs and the actual digital voice inputs. Now, with this approach the system will find the similarities that are present for good range of speakers and thus the system need not to be trained by the user before using it for voice detection purpose.There are many ways of characterizing an audio signal. Mostly, the audio signals are categorized into two domains: time-domain and frequency-domain features.

Feature analysis is done on each clip by calculating a feature vector for each clip. These features are calculated on the basis of frame level features. The frame level features are computed by overlapping short intervals known as frames. feature extraction technique can recognize a speech that is spoken in different accents and varying speeds of speech delivery, pitch and volume.

Pre-emphasis

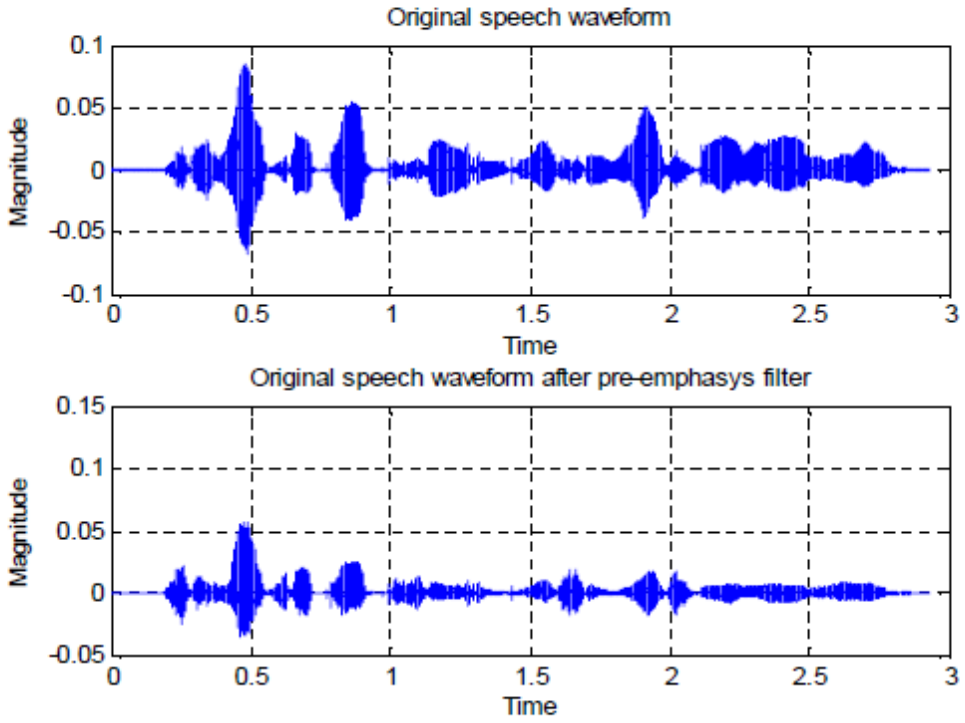
In speech processing, the original signal usually has too much lower frequency energy, and processing the signal to emphasize higher frequency energy is necessary. To perform pre-emphasis, we choose some value α between 0.95 and 0.97 Then each value in the signal is re-evaluated using this formula: y[n] = x[n] - α*x[n-1]. This is apparently a first order high pass filter.

$$H_{preem}(z) = 1 - a_{preem}z^{-1}$$



Pre-emphasis Filter, a=0.97.

In order to flatten speech spectrum, a pre-emphasis filter is used before spectral analysis. Its aim is to compensate the high frequency part of the speech signal that was suppressed during the human sound production mechanism.



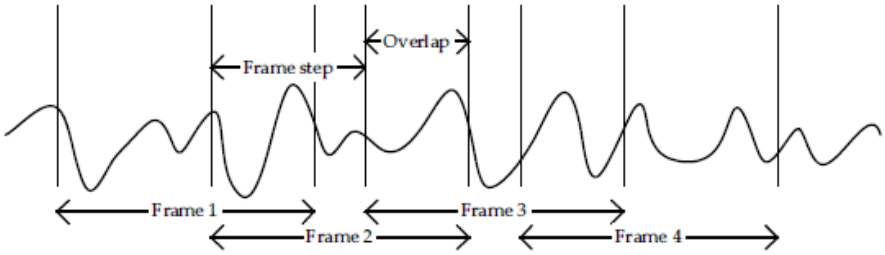
Original speech waveform and original speech waveform after the pre-emphasis filter with coefficient equal to 0.97

Framing

In speech processing it is often advantageous to divide the signal into frames to achieve stationarity.

Each frame shares the first part with the previous frame and the last part with the next frame. The time frame step tf s indicates how long time there is between the start time of each frame. The overlap to is defined as the time from a new frame starts until the current stops. From this follows that the frame length tf l is:

tf l = tf s +to

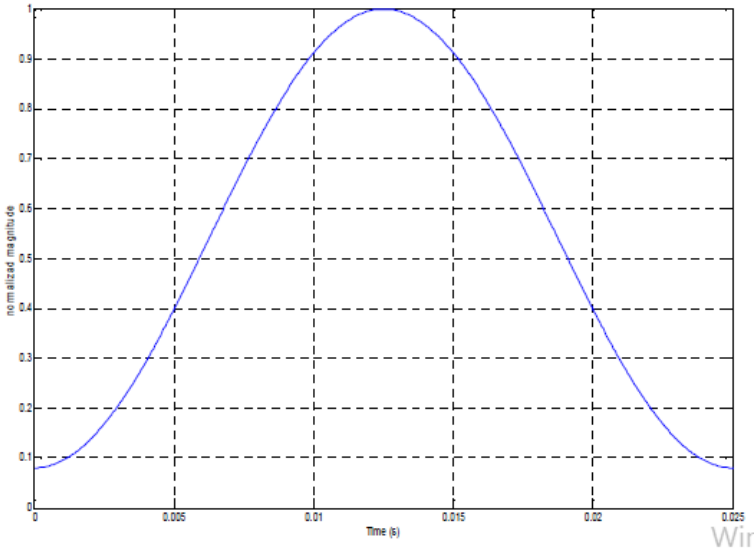


the speech signal is divided into a sequence of frames where each frame can be analyzed independently and represented by a single feature vector. Since each frame is supposed to have stationary behaviour, a compromise, in order to make the frame blocking, is to use a 20-25 ms window applied at 10 ms intervals (frame rate of 100 frames/s and overlap between adjacent windows of about 50%)

Windowing

In order to reduce the discontinuities of the speech signal at the edges of each frame, a tapered window is applied to each one. The most common used window is Hamming window.

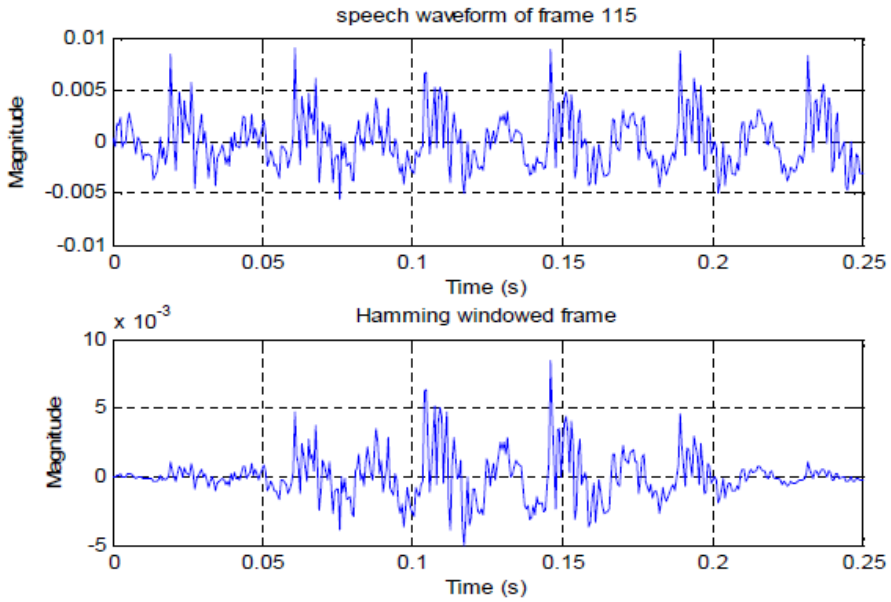
$$w(n) = 0.54 - 0.46\cos\left(\frac{2\pi(n-1)}{\mathrm{N}-1}\right)$$



25ms Hamming window (fs=16Khz)

Speech is non-stationary signal where properties change quite rapidly over time. the properties of the speech remain invariant for a short period of time (5-100 ms). Thus for a short window of time, traditional signal processing methods can be applied relatively successfull.

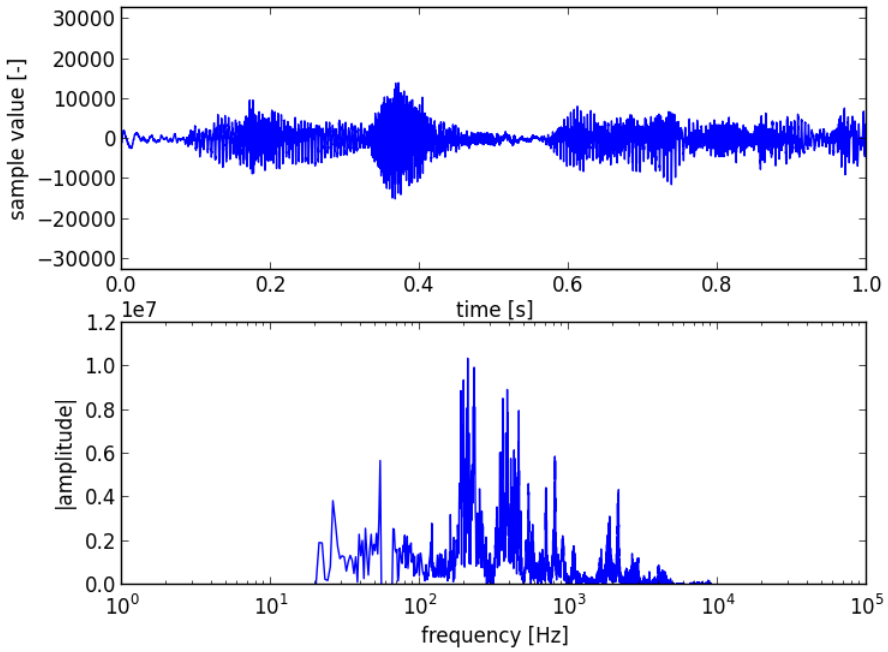
In speech processing the shape of the window function is not that crucial but usually some soft window like Hanning, Hamming, triangle, half parallerogram, not with right angles. The reason is same as in filter design, sideband lobes as substantially smaller than in a rectangular window.



Effect of multiplying one speech frame by a Hamming window

Fast Fourier Transform(FFT)

A fast Fourier transform (FFT) is an algorithm that computes the discrete Fourier transform (DFT) of a sequence, or its inverse (IDFT). Fourier analysis converts a signal from its original domain (often time or space) to a representation in the frequency domain and vice versa.



Fast Fourier Transform (FFT) is the traditional technique to analyze frequency spectrum of the signal in speech recognition. Speech recognition operation requires heavy computation due to large samples per window. The preliminary experimental results show that DTT has the potential to be a simpler and faster transformation for speech recognition.

FFT is an efficient algorithm that can perform Discrete Fourier Transform (DFT). FFT is applied in order to convert time domain signals into the frequency domain. The FFT takes advantage of the symmetry and periodicity properties of the Fourier Transform to reduce computation time. In this process, the transform is partitioned into a sequence of reduced-length transforms that is collectively performed with reduced computation.

$$X_k = \sum_{n=0}^{N-1} x_n e^{-i 2\pi k n / N} \qquad k = 0, \ldots, N-1.$$

Evaluating this definition directly requires O(n^2) operations: there are N outputs X_k, and each output requires a sum of N terms. An FFT is any method to compute the same results in O(nlog(n)) operations. All known FFT algorithms require nlog(n) operations, although there is no known proof that a lower complexity score is impossible.

To illustrate the savings of an FFT, consider the count of complex multiplications and additions for N=4096 data points. Evaluating the DFT's sums directly involves N^2 complex multiplications and N(N−1) complex additions, of which O(n) operations can be saved by eliminating trivial operations such as multiplications by 1, leaving about 30 million operations. On the other hand, the radix-2 Cooley–Tukey algorithm, for N a power of 2, can compute the same result with only (N/2)$\log_2$(N) complex multiplications (again, ignoring simplifications of multiplications by 1 and similar) and N $\log_2$(N) complex additions, in total about 30,000 operations - a thousand times less than with direct evaluation.

Mel Frequency Cepstrum Coefficient(MFCC)

The MFCC is a representation of the speech signal defined as the real cepstrum of a windowed short-time signal derived from the FFT of that signal which, is first subjected to a log-based transform of the frequency axis (mel-frequency scale), and then decorrelated using a modified Discrete Cosine Transform (DCT-II).

Steps in Converting from frequency domain to time domain

This figure illustrates the complete process to extract the MFFC vectors from the speechsignal. It is to be emphasized that the process of MFCC extraction is applied over eachframe of speech signal independently.

Mel Frequency

The Mel scale relates perceived frequency, or pitch, of a pure tone to its actual measured frequency. Humans are much better at discerning small changes in pitch at low frequencies than they are at high frequencies. Incorporating this scale makes our features match more closely what humans hear.

The formula for converting from frequency to Mel scale is:

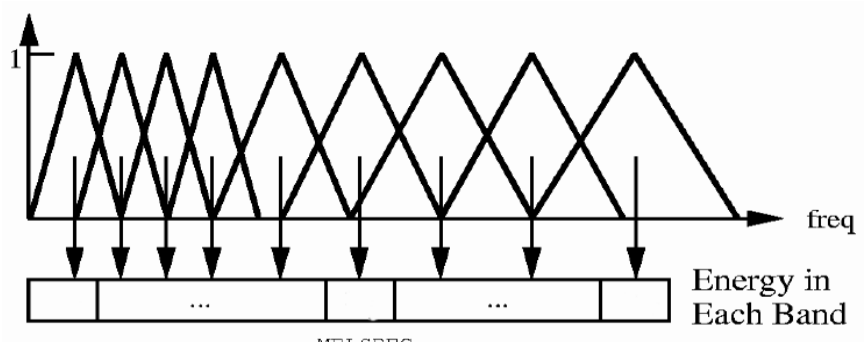
$$M(f) = 1125 \ln(1 + f/700) \qquad (1)$$

To go from Mels back to frequency:

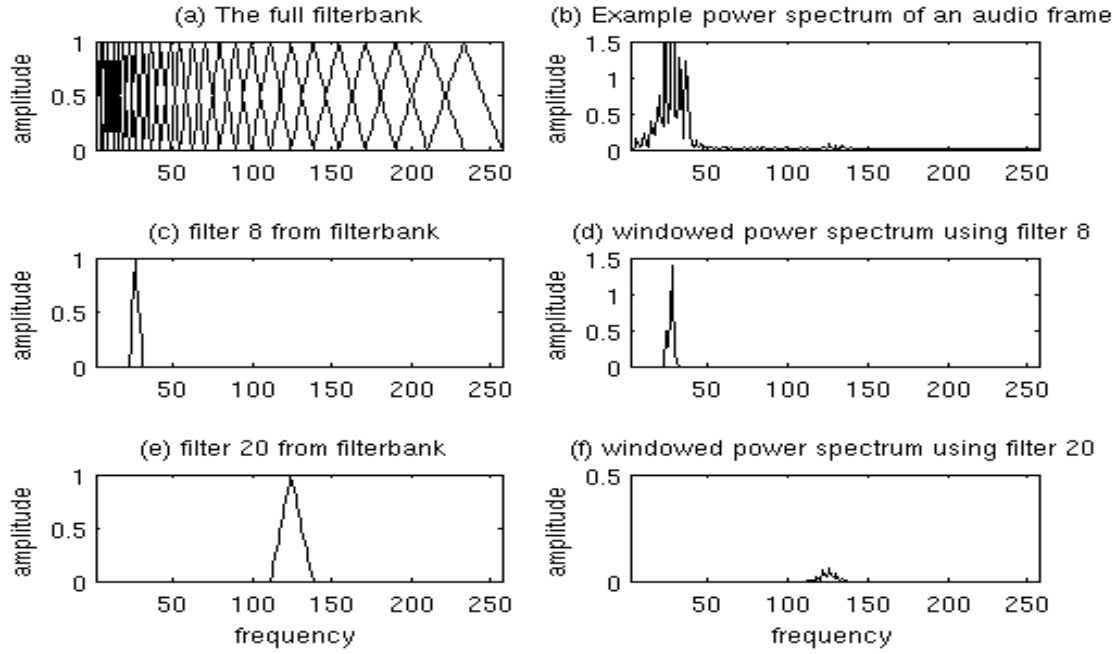
$$M^{-1}(m) = 700(\exp(m/1125) - 1) \qquad (2)$$

Mel Filter Bank Processing

The frequencies range in FFT spectrum is very wide and voice signal does not follow the linear scale. The bank of filters according to Mel scale as shown in figure is then performed.



This figure shows a set of triangular filters that are used to compute a weighted sum of filter spectral components so that the output of process approximates to a Mel scale. Each filter's magnitude frequency response is triangular in shape and equal to unity at the centre frequency and decrease linearly to zero at centre frequency of two adjacent filters . Then, each filter output is the sum of its filtered spectral components. After that the following equation is used to compute the Mel for given frequency f in HZ:

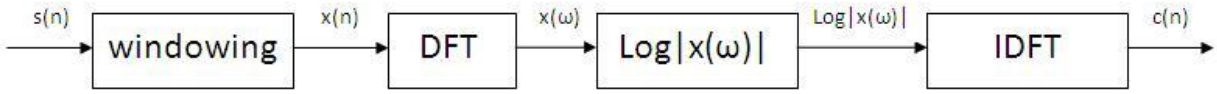


Cepstral Analysis

A signal comming out from a system is due to the input excitation and also the response of the system. From the signal processing point of view, the output of a system can be treated

as the convolution of the input excitation with the system response. We need each of the components separately for study and processing.

Speech is composed of excitation source and vocal tract system components. In order to analyze and model the excitation and system components of the speech independently and also use that in various speech processing applications, these two components have to be separated from the speech. The objective of cepstral analysis is to separate the speech into its source and system components without any a priori knowledge about source and system.



According to the source filter theory of speech production, voiced sounds are produced by exciting the time varying system characteristics with periodic impulse sequence and unvoiced sounds are produced by exciting the time varying system with a random noise sequence. The resulting speech can be considered as the convolution of respective excitation sequence and vocal tract filter characteristics. If e(n) is the excitation sequence and h(n) is the vocal tract filter sequence, then the speech sequence s(n) can be expressed as follows:

$$s(n) = e(n) * h(n) \qquad (1)$$

This can be represented in frequency domain as,

$$S(\omega) = E(\omega).H(\omega) \qquad (2)$$

The Eqn. (2) indicates that the multiplication of excitation and system components in the frequency domain for the convolved sequence of the same in the time domain. The speech sequence has to be deconvolved into the excitation and vocal tract components in the time domain. For this, multiplication of the two components in the frequency domain has to be converted to a linear combination of the two components. For this purpose cepstral analysis is used for transforming the multiplied source and system components in the frequency domain to linear combination of the two components in the cepstral domain.

From the Eqn. (2) the magnitude spectrum of given speech sequence can be represented as,

$$|S(\omega)| = |E(\omega)|.|H(\omega)| \qquad (3)$$

To linearly combine the E(ω) and H(ω) in the frequency domain, logarithmic representation is used. So the logarithmic representation of Eqn. (3) will be,

$$log|S(\omega)| = log|E(\omega)| + log|H(\omega)| \qquad (4)$$

As indicated in Eqn. (4), the log operation transforms the magnitude speech spectrum where the excitation component and vocal tract component are multiplied, to a linear

combination (summation) of these components i.e. log operation converted the "*" operation into "+" operation in the frequency domain. The separation can be done by taking the inverse discrete fourier transform (IDFT) of the linearly combined log spectra of excitation and vocal tract system components. It should be noted that IDFT of linear spectra transforms back to the time domain but the IDFT of log spectra transforms to quefrency domain or the cepstral domain which is similar to time domain.

$$c(n) = IDFT(log|S(\omega)|) = IDFT(log|E(\omega)| + log|H(\omega)|) \quad (5)$$

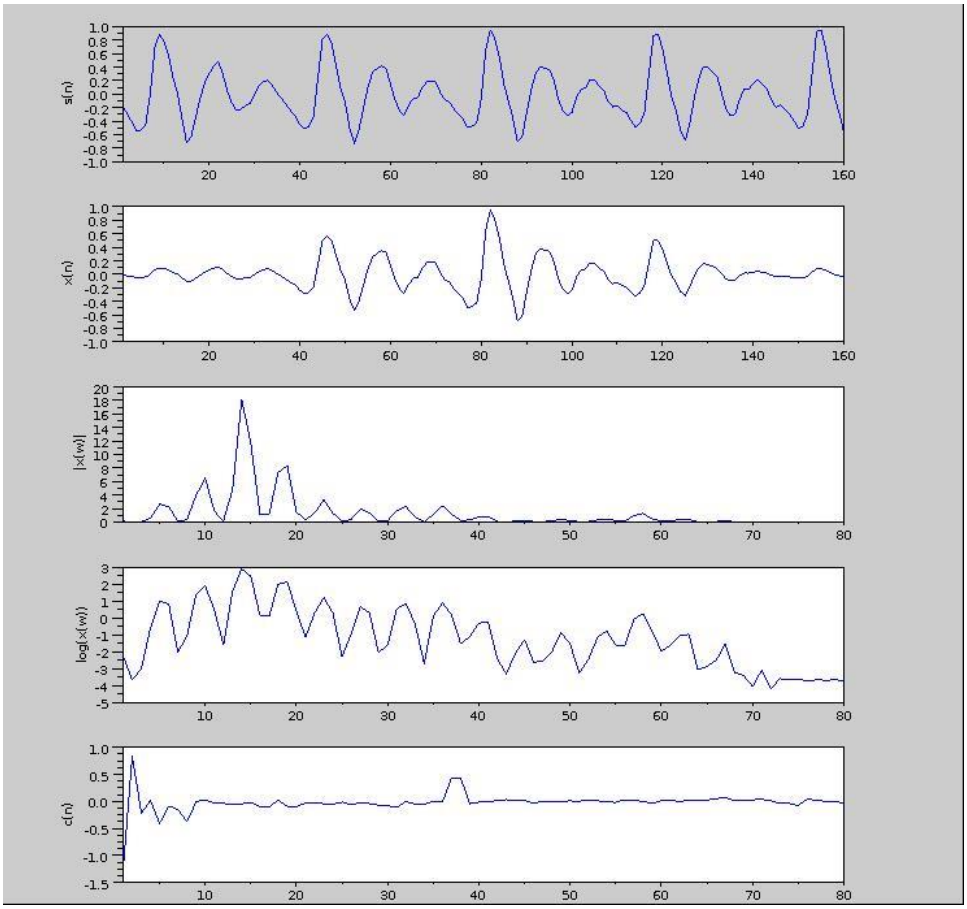


Figure: 20 ms voiced speech segment and its cepstrum

Discrete Cosine Transform

A discrete cosine transform (DCT) expresses a finite sequence of data points in terms of a sum of cosine functions oscillating at different frequencies. DCTs are important to numerous applications in science and engineering, from lossy compression of audio (e.g. MP3) and images (e.g. JPEG) (where small high-frequency components can be discarded), to spectral methods for the numerical solution of partial differential equations.

$$X_k = \sum_{n=0}^{N-1} x_n \cos\left[\frac{\pi}{N}\left(n+\frac{1}{2}\right)k\right] \qquad k = 0, \ldots, N-1.$$

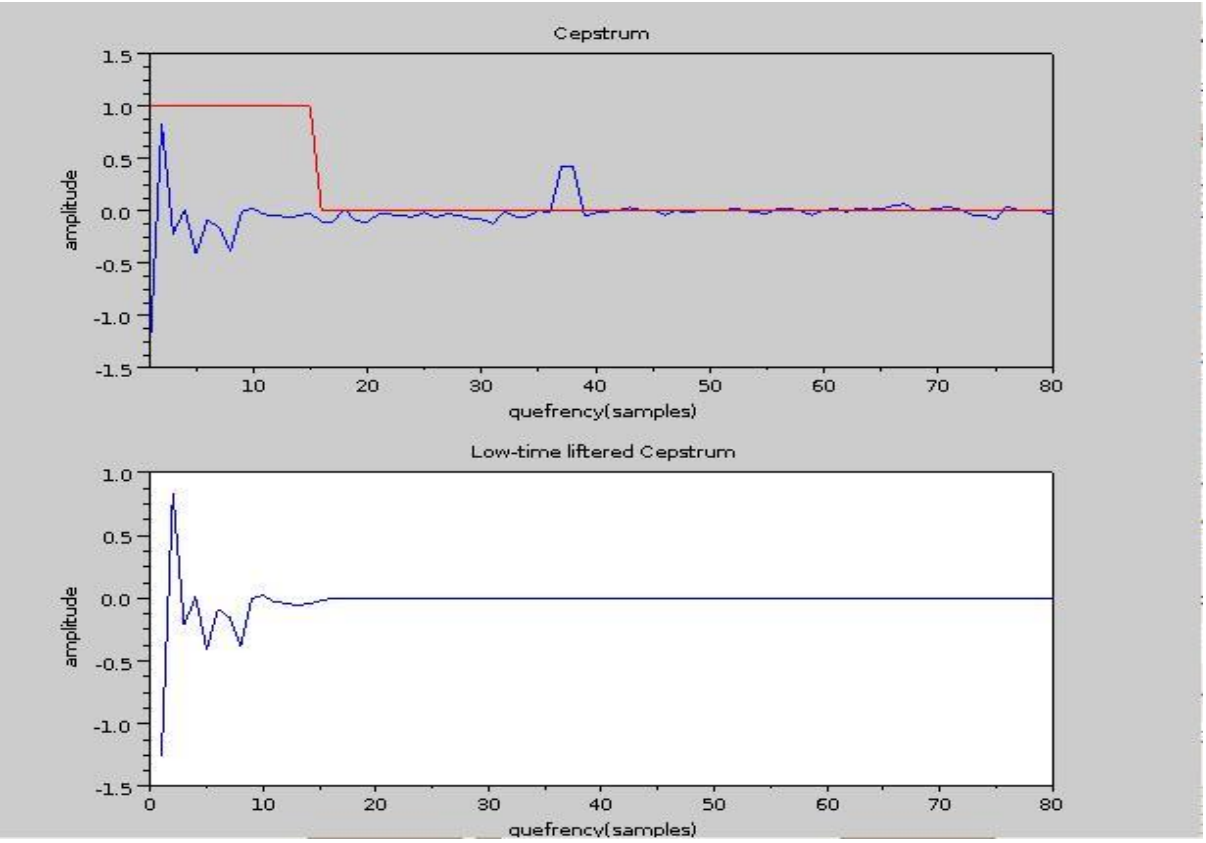
In particular, a DCT is a Fourier-related transform similar to the discrete Fourier transform (DFT), but using only real numbers.

This is the process to convert the log Mel spectrum into time domain using Discrete Cosine Transform (DCT). The result of the conversion is called Mel Frequency Cepstrum Coefficient.

Liftering

Liftering operation is similar to filtering operation in the frequency domain where a desired quefrency region for analysis is selected by multiplying the whole cepstrum by a rectangular window at the desired position. There are two types of liftering performed, low-time liftering and high-time liftering. Low-time liftering operation is performed to extract the vocal tract characteristics in the quefrency domain and high-time liftering is performed to get the excitation characteristics of the analysis speech frame.

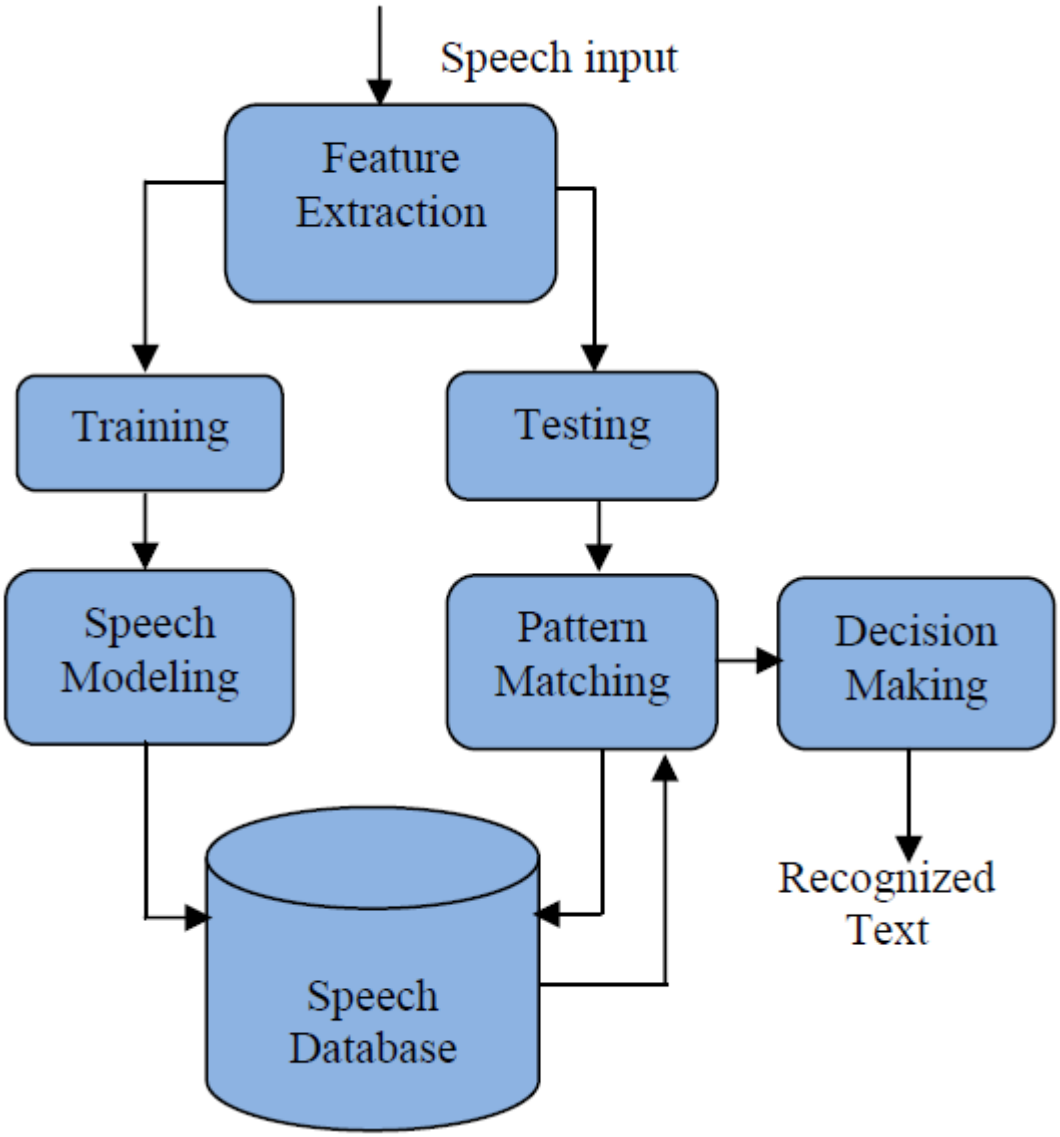
Low-time liftering is used for estimating slow varying vocal tract characteristics from the computed cepstrum of the given speech sequence.



The cepstral liftering is achieved by applying multiplicative weights {wi} to cepstral coefficients. Because of this, liftering of cepstral coeffcients has no effect in the recognition process when used with continuous observation Gaussian density HMMs. Since the multiplicative weighting of cepstral coeffcients is equivalent to convolution in spectral domain, liftering of filter bank energies can be used with the continuous Gaussian density HMMs with positive effect on recognition performance.

Feature Matching

In speech recognition system, Feature Extraction plays an important role to separate one speech from other. The main focus of feature extractor is to keep the relevant information and discard irrelevant one from the speech. Matching is performed by comparing a voice sample with the sample in the database.



There are two essential steps involved in pattern recognition approach, pattern training and pattern comparison. Using a well formulated mathematical framework and initiates consistent speech pattern representation for reliable pattern comparison. A set of labeled training samples through formal training algorithm is essential feature of this approach. In this, there exist two methods: Template base approach and stochastic approach. It is more suitable approach to speech recognition as it uses probabilistic models to deal with undetermined or incomplete information.There exist many methods in this approach like HMM, SVM, DTW, VQ etc, among these Hidden Markov Model is most popular stochastic approach today.

Hidden Markov Model (HMM)

Modern general purpose speech recognition systems are based on Hidden Markov Models. These are statistical models that output a sequence of symbols or quantities. HMMs are used in speech recognition because a speech signal can be viewed as a piecewise stationary signal or a short-time stationary signal. In a short time-scale (e.g., 10 milliseconds), speech can be approximated as a stationary process. Speech can be thought of as a Markov model for many stochastic purposes.

Systems can be trained automatically and are simple and computationally feasible to use. In speech recognition, the Hidden Markov Model would output a sequence of n-dimensional real-valued vectors (with n being a small integer, such as 10), outputting one of these every 10 milliseconds.The vectors would consist of cepstral coefficients, which are obtained by taking a Fourier transform of a short time window of speech and decorrelating the spectrum using a cosine transform, then taking the first (most significant) coefficients. The Hidden Markov Model will tend to have in each state a statistical distribution that is a mixture of diagonal covariance Gaussians, which will give a likelihood for each observed vector. Each word, or (for more general speech recognition systems), each phoneme, will have a different output distribution; a Hidden Markov Model for a sequence of words or phonemes is made by concatenating the individual trained hidden Markov models for the separate words and phonemes.

Hidden Markov Model (HMM) is a statistical Markov model in which the system being modeled is assumed to be a Markov process with unobserved (i.e. hidden) states.

In simpler Markov models (like a Markov chain), the state is directly visible to the observer, and therefore the state transition probabilities are the only parameters, while in the hidden Markov model, the state is not directly visible, but the output (in the form of data or "token" in the following), dependent on the state, is visible. Each state has a probability distribution over the possible output tokens. Therefore, the sequence of tokens generated by an HMM gives some information about the sequence of states; this is also known as pattern theory, a topic of grammar induction.