Multimedia Information and Signal Processing

Lecture 4: Feature extraction from multimedia signals

Zheng-Hua Tan
Dept. of Electronic Systems, Aalborg Univ., Denmark
zt@es.aau.dk, http://kom.aau.dk/~zt

Course outline

1. Introduction
2. Acquisition and representation of multimedia signals
3. Feature extraction from speech, music, images, etc.
4. Bayes decision theory: Bayes rule, loss function
5. Parametric and nonparametric methods
6. Supervised learning (of classification and regression functions): K-nearest neighbors, decision trees, linear regression, linear discriminant analysis, multilayer perceptrons
7. Unsupervised learning (for clustering, density estimation and dimensionality reduction): K-means, Gaussian mixture model, principal component analysis
8. Model selection: bias and variance, boosting and cross-validation
9. Applications

What is feature extraction?

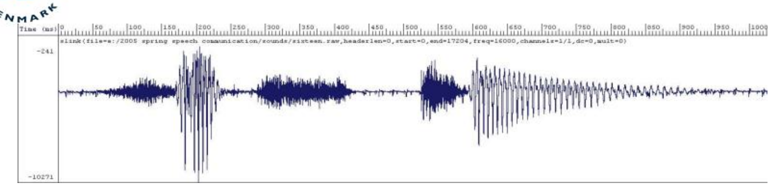
- A special form of dimensionality reduction, used when the input data is
  - Too large to be stored or processed
  - Redundant (much data, but not much information)

- Data is transformed into a compact representation – a set of features.

Lecture outline

- Sound and speech
  - Short-time speech analysis
  - Time-domain processing
  - Frequency-domain (spectral) processing
  - Mel-frequecy cepstral coefficients (MFCC)

Properties of speech signals



Speech is a time-varying signal:

Short-time processing solution

Assuming that speech has non-time-varying properties (fixed excitation and vocal tract) within short intervals →

Processing short segments (frames) of the speech signal each time

$$f_x(n,m) = x(m)w(n-m)$$

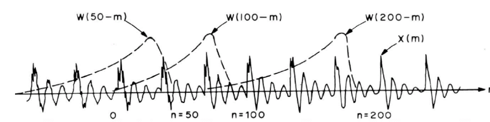
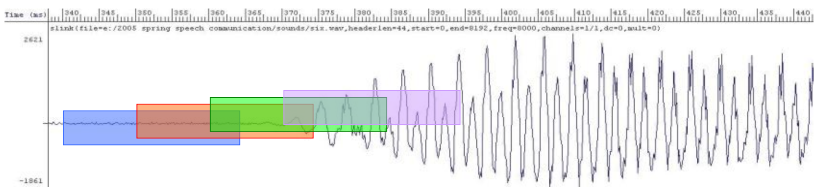


Fig. 6.1 Sketches of $x(m)$ and $w(n-m)$ for several values of n.

Frame-by-frame processing

- frames (segments) often overlap one another



- The frame-based analysis yields a time-varying sequence as a new representation of the speech signal
  - samples at 8000/sec → vectors at 100/sec

Time-domain parameters

- Short-time energy
- Short-time zero crossing rate
- Short-time autocorrelation
- Short-time average magnitude difference

Short-time energy

- The long term energy definition is not useful for time-varying signals

$$E = \sum_{m=-\infty}^{\infty} x^2(m)$$

- Short-time energy of weighted signal around n is defined as

$$E_n = \sum_{m=-\infty}^{\infty} [x(m)w(n-m)]^2$$

Examples of short-time energy

- It can be used to detection voiced/unvoiced/silence
  - Effects of window type

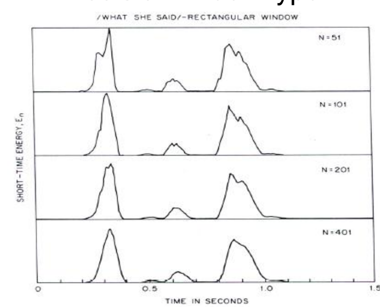


Fig. 4.6 Short-time energy functions for rectangular windows of various lengths.

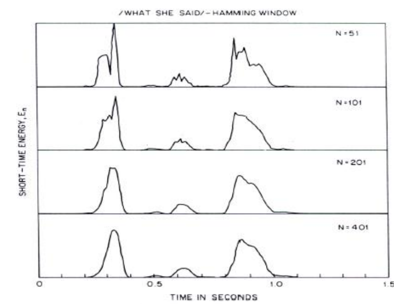


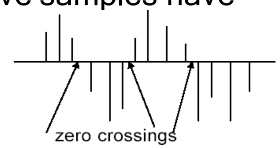
Fig. 4.7 Short-time energy functions for Hamming windows of various lengths.

Uttered by a male speaker.

Two plots converge as N increases.

Short-time average zero-crossing rate

- A zero-crossing occurs if successive samples have different algebraic signs.
- It is a measure of the frequency.
- Definition



$$Z_n = \sum_{m=-\infty}^{\infty} |\,\mathrm{sgn}[x(m)] - \mathrm{sgn}[x(m-1)]\,|\, w(n-m)$$

where $$\mathrm{sgn}[x(n)] = \begin{cases} 1 & x(n) \geq 0 \\ -1 & x(n) < 0 \end{cases}$$

$$w(n) = \begin{cases} \dfrac{1}{2N} & 0 \leq n \leq N-1 \\ 0 & otherwise \end{cases}$$

Example of zero-crossing rate

- Although the zero-crossing rate varies considerably, the voiced and unvoiced regions are quite prominent.

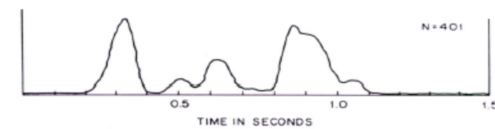


Fig. 4.9 Average magnitude functions for Hamming windows

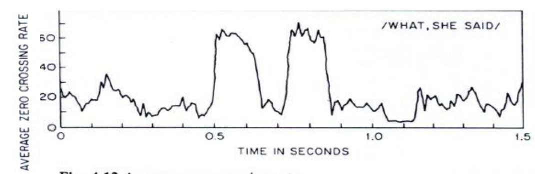


Fig. 4.12 Average zero-crossing rate

Short-time autocorrelation function

- The autocorrelation function

$$\phi(k) = \sum_{m=-\infty}^{\infty} x(m)x(m+k)$$

- The short-time autocorrelation function

$$R_n(k) = \sum_{m=-\infty}^{\infty} x(m)w(n-m)x(m+k)w(n-k-m)$$

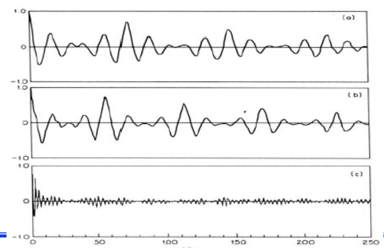


Fig. 4.24 Autocorrelation function for (a) and (b) voiced speech; and (c) unvoiced speech, using a rectangular window with $N = 401$.

Short-time Fourier transform

- It is motivated by the need for a spectral representation to reflect the time-varying properties of the speech waveform

$$X_n(e^{jw}) = \sum_{m=-\infty}^{+\infty} w[n-m]x[m]e^{-jwm}$$

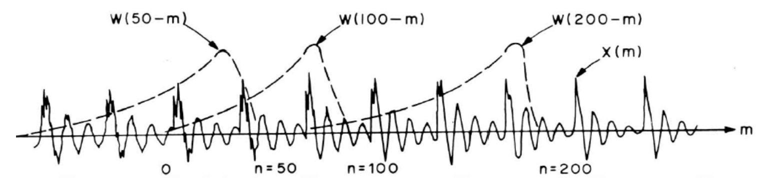
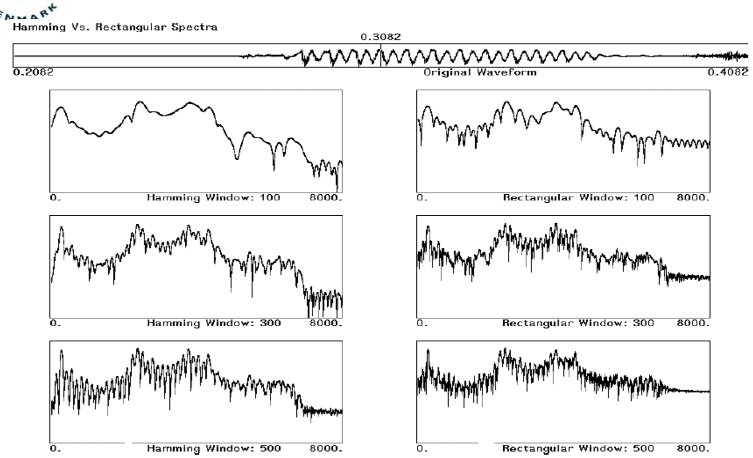


Fig. 6.1 Sketches of $x(m)$ and $w(n-m)$ for several values of n.

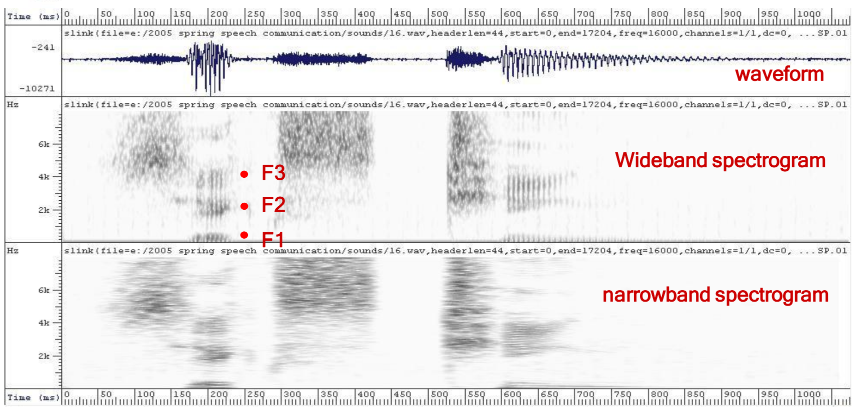
Spectra



Spectrogram

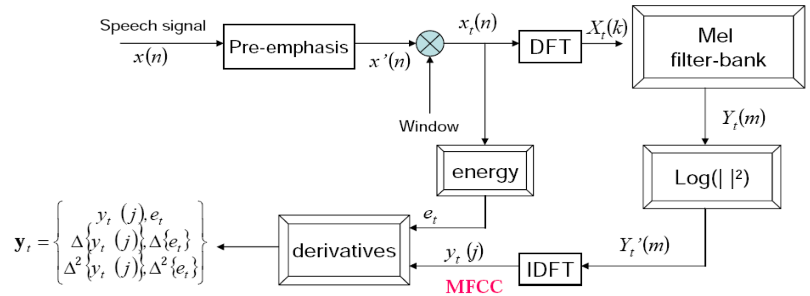
- two-dimensional waveform (amplitude/time) is converted into a three-dimensional pattern (amplitude/frequency/time)
- Wideband spectrogram: analyzed on 15ms sections of waveform with a step of 1ms
  - voiced regions with vertical striations due to the periodicity of the time waveform (each vertical line represents a pulse of vocal folds) while unvoiced regions are solid/random, or ‘snowy’
- Narrowband spectrogram: on 50ms
  - pitch for voiced intervals in horizontal lines

Wide- and narrow-band spectrograms



MFCC

- Mel-Frequency Cepstral Coefficient (MFCC)
  - Most widely used spectral representation in ASR

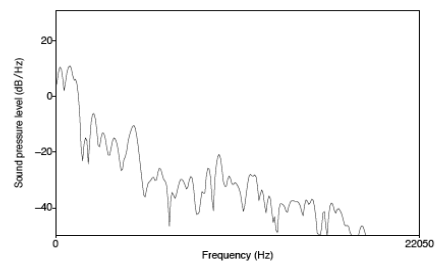


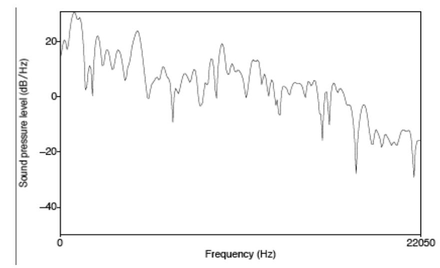
Pre-Emphasis

- Pre-emphasis: boosting the energy in the high frequencies
- Q: Why do this?
- A: The spectrum for voiced segments has more energy at lower frequencies than higher frequencies.
  - This is called **spectral tilt**
  - Spectral tilt is caused by the nature of the glottal pulse
- Boosting high-frequency energy gives more info to Acoustic Model
  - Improves phone recognition performance

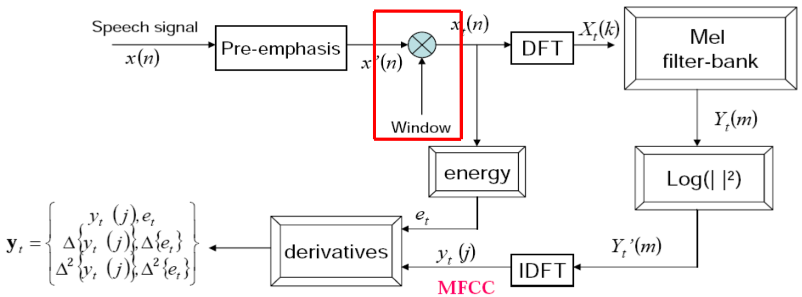
Example of pre-emphasis

- Before and after pre-emphasis
  - Spectral slice from the vowel [aa]

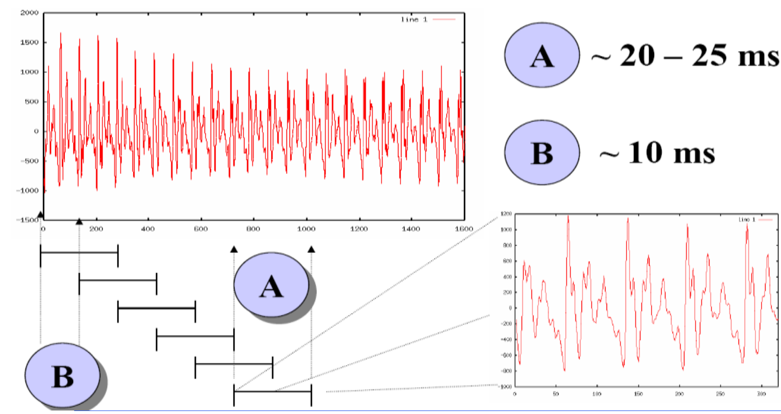




MFCC



Windowing



Windowing

- Why divide speech signal into successive overlapping frames?
  - Speech is not a stationary signal; we want information about a small enough region that the spectral information is a useful cue.
- Frames
  - Frame size: typically, 10-25ms
  - Frame shift: the length of time between successive frames, typically, 5-10ms

Common window shapes

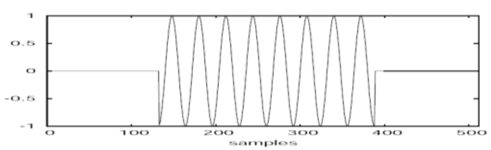
- Rectangular window:

$$w[n] = \begin{cases} 1 & 0 \leq n \leq L-1 \\ 0 & \text{otherwise} \end{cases}$$

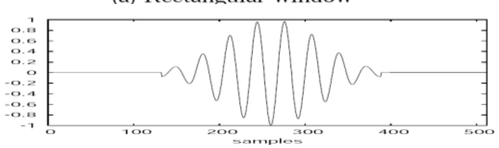
- Hamming window

$$w[n] = \begin{cases} 0.54 - 0.46 \cos\left(\frac{2\pi n}{L-1}\right) & 0 \leq n \leq L-1 \\ 0 & \text{otherwise} \end{cases}$$

Window in time domain

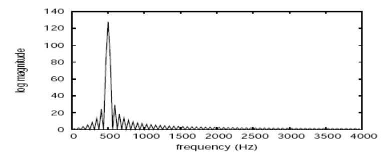


(a) Rectangular window

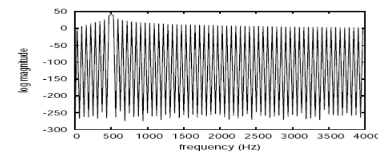


(c) Hamming window

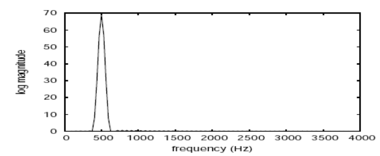
Window in the frequency domain



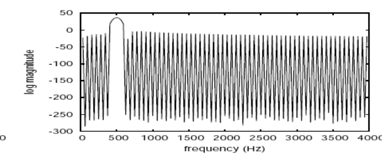
(a) Rectangular window



(b) Rectangular window

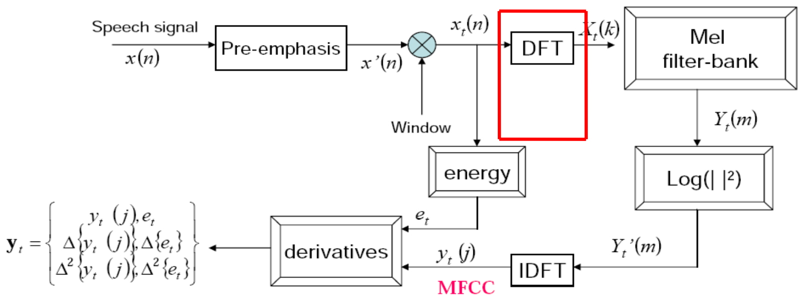


(e) Hamming window



(f) Hamming window

MFCC



Discrete Fourier Transform

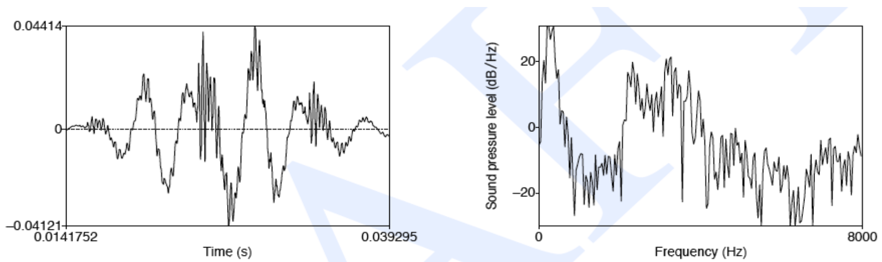
- Input:
  - Windowed signal x[n]…x[m]
- Output:
  - For each of N discrete frequency bands
  - A complex number X[k] representing magnidue and phase of that frequency component in the original signal
- Discrete Fourier Transform (DFT)

$$X[k] = \sum_{n=0}^{N-1} x[n] e^{-j2\frac{\pi}{N}kn}$$

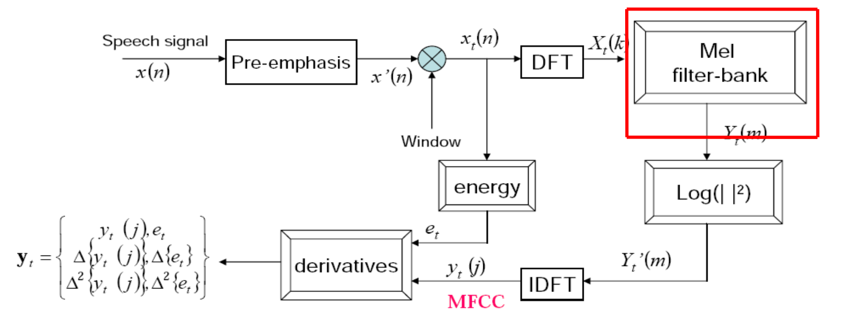
- Standard algorithm for computing DFT:
  - Fast Fourier Transform (FFT) with complexity N*log(N)
  - In general, choose N=512 or 1024

Discrete Fourier Transform computing a spectrum

- A 24 ms Hamming-windowed signal
  - And its spectrum as computed by DFT (plus other smoothing)

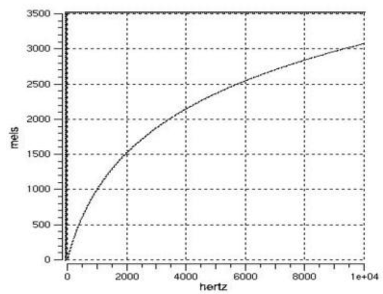


MFCC



Mel-scale

- Human hearing is not equally sensitive to all frequency bands
- Less sensitive at higher frequencies, roughly > 1000 Hz
- I.e. human perception of frequency is non-linear:



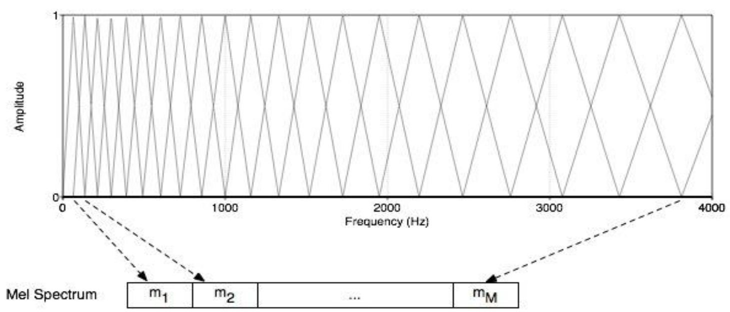
Mel-scale

- A **mel** is a unit of pitch
  - Definition:
    - Pairs of sounds perceptually equidistant in pitch
      - Are separated by an equal number of mels:
- Mel-scale is approximately linear below 1 kHz and logarithmic above 1 kHz
- Definition:

$$Mel(f) = 2595 \log_{10} \left(1 + \frac{f}{700}\right)$$

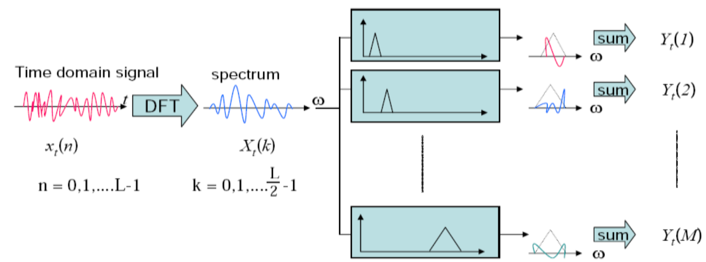
Mel Filter Bank Processing

- Mel Filter bank
  - Uniformly spaced before 1 kHz
  - logarithmic scale after 1 kHz

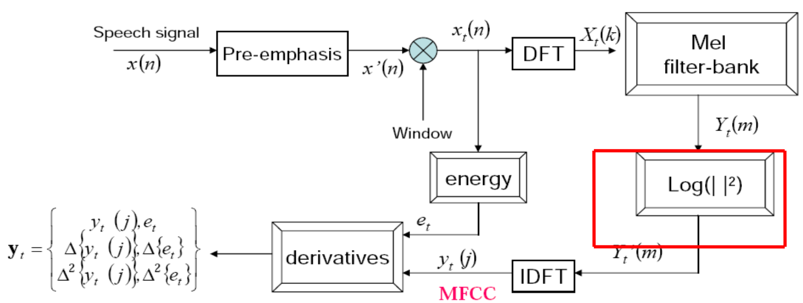


Mel-filter Bank Processing

- Apply the bank of filters according Mel scale to the spectrum
- Each filter output is the sum of its filtered spectral components

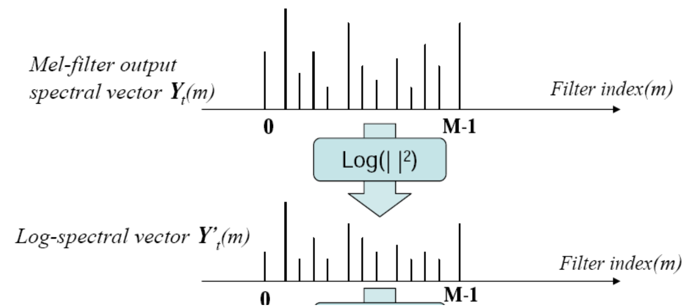


MFCC



Log energy computation

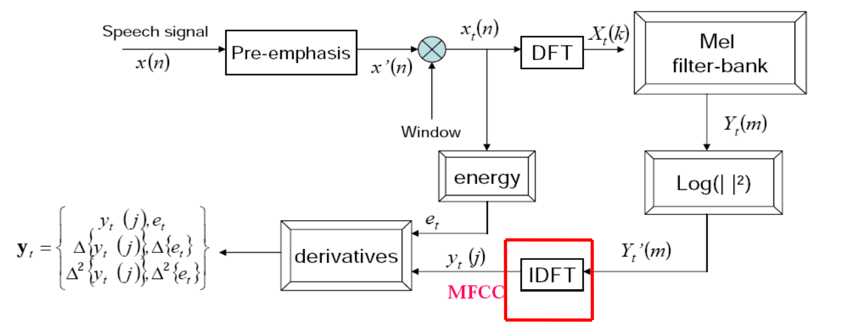
- Compute the logarithm of the square magnitude of the output of Mel-filter bank



Log energy computation

- Why log energy?
- Logarithm compresses dynamic range of values
  - Human response to signal level is logarithmic
  - humans less sensitive to slight differences in amplitude at high amplitudes than low amplitudes
- Makes frequency estimates less sensitive to slight variations in input (power variation due to speaker's mouth moving closer to mike)
- Phase information not helpful in speech

MFCC



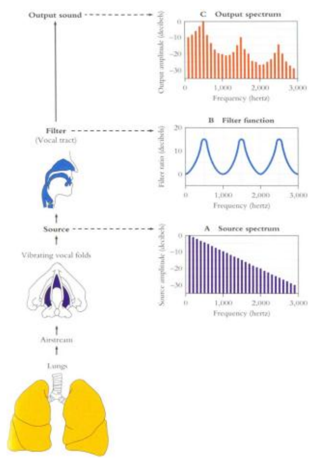
The Cepstrum

- One way to think about this
  - Separating the **source** and **filter**
  - Speech waveform is created by
    - A glottal source waveform
    - Passes through a vocal tract which because of its shape has a particular filtering characteristic
- Articulatory facts:
  - The vocal cord vibrations create harmonics
  - The mouth is a filter
  - Depending on shape of oral cavity, some harmonics are attenuated more than others

Vocal Fold Vibration

UCLA Phonetics Lab Demo

George Miller figure

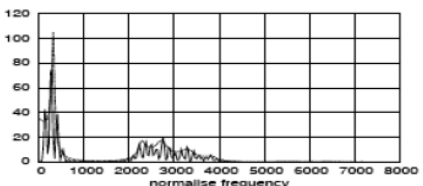


We care about the filter not the source

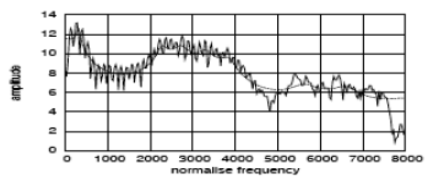
- Most characteristics of the source
  - F0
  - Details of glottal pulse
- Don't matter for phone detection
- What we care about is the **filter**
  - The exact position of the articulators in the oral tract
- So we want a way to separate these
  - And use only the filter function

The Cepstrum

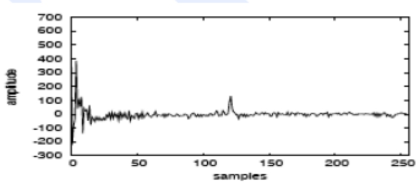
- The spectrum of the log of the spectrum



Spectrum

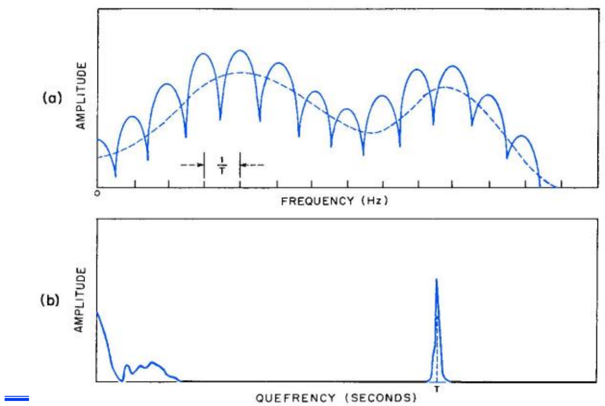


Log spectrum



Spectrum of log spectrum

Thinking about the Cepstrum



Mel Frequency cepstrum

- The cepstrum requires Fourier analysis
- But we're going from frequency space back to time
- So we actually apply inverse DFT

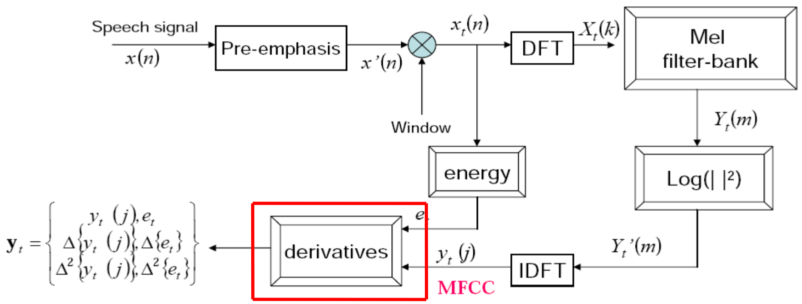
$$y_t[k] = \sum_{m=1}^{M} \log(|Y_t(m)|) \cos(k(m-0.5)\frac{\pi}{M}),\ \text{k=0,...,J}$$

- Details for signal processing gurus: Since the log power spectrum is real and symmetric, inverse DFT reduces to a Discrete Cosine Transform (DCT)

Another advantage of the Cepstrum

- DCT produces highly **uncorrelated** features
- We'll see when we get to acoustic modeling that these will be much easier to model than the spectrum
  - Simply modelled by linear combinations of Gaussian density functions with diagonal covariance matrices
- In general we'll just use the first 12 cepstral coefficients

MFCC

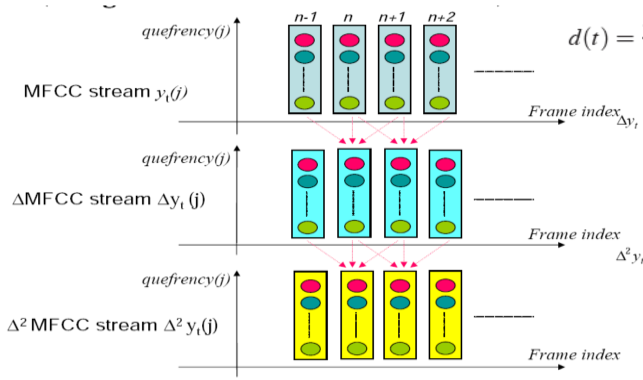


Dynamic Cepstral Coefficient

- The cepstral coefficients do not capture energy
- So we might (be careful) add an energy feature: $Energy = \sum_{t=t_1}^{t_2} x^2[t]$
- Also, we know that speech signal is not constant (slope of formants, change from stop burst to release).
- So we want to add the changes in features (the slopes).
- We call these **delta** features
- We also add **double-delta** acceleration features

Delta and double-delta

- Derivative: in order to obtain temporal information



$$d(t) = \frac{c(t+1) - c(t-1)}{2}$$

$$\Delta y_t(j) = \frac{\sum_{m=-p}^{p} m \bullet y_{t-m}(j)}{\sum_{m=-p}^{p} m^2}$$

$$\Delta^2 y_t(j) = \frac{\sum_{m=-p}^{p} m \bullet \Delta y_{t-m}(j)}{\sum_{m=-p}^{p} m^2}$$

Typical MFCC features

- Window size: 25ms
- Window shift: 10ms
- Pre-emphasis coefficient: 0.97
- MFCC:
  - 12 MFCC (mel frequency cepstral coefficients)
  - (1 energy feature)
  - 12 delta MFCC features
  - 12 double-delta MFCC features
  - 1 delta energy feature
  - 1 double-delta energy feature
- Total 38-39-dimensional features

Why is MFCC so popular?

- Efficient to compute
- Incorporates a perceptual Mel frequency scale
- Separates the source and filter
- IDFT(DCT) decorrelates the features
  - Improves diagonal assumption in e.g. HMM/HMM modelling

That's it for today!

- You learned:
  - Preprocessing of speech both in time- and frequency domain
  - Details of Mel-Frequency Cepstral Coefficients (MFCC)