T-61.184
Automatic Speech Recognition: From Theory to Practice

http://www.cis.hut.fi/Opinnot/T-61.184/
September 27, 2004

Prof. Bryan Pellom
Department of Computer Science
Center for Spoken Language Research
University of Colorado

pellom@cslr.colorado.edu

Homework Exercises 1 & 2

- **Due date extended to October 4th (latest)**
  - I understand that some people have computing issues and need more time
- **Homework #2 Due October 4th (hard deadline)**
  - Does not involve use of computers
  - Same login/password as HW1
  - There will be no time extensions on HW2!

Homework #1: Recap

- **Fundamental concepts**
  - Notion of Training, Development and Final Test sets
  - Feature extraction
  - Viterbi alignment of training data
  - Estimation of HMM model parameters from audio and Language Model from text data
  - How to measure the word error rate of the final system
- **Advanced concepts**
  - Vocal Tract Length Normalization
  - Speaker Adaptation
  - Speaker Adaptive Training

Expected Outcomes

- **Topics we'll be talking about this term,**
  - ❑ Feature Extraction
  - ❑ HMM Modeling and data alignment
  - ❑ Computing Word Error Rates
  - ❑ Estimation of Statistical Language Models
  - ❑ Speaker Adaptation (MLLR, VTLN)
  - ❑ Speaker Adaptive Training

- **You might not understand all the concepts in HW1, but hopefully were able to walk through each of the steps. As the term progresses, the items in the first homework will become more clear.**

Today’s Outline

- Consider ear physiology
- Consider evidence from psycho-acoustic experiments
- Review current methods for speech recognition feature extraction
- Some considerations of what (possibly) we are doing wrong in the ASR field

Ear Physiology

- **Outer Ear:** **2.5 cm long**
  - Pinna
  - Auditory Canal
  - Tympanic Membrane (Eardrum)
- **Middle Ear**
  - Ossicles
    - 3 bones: Malleus, Incus, Stapes
  - Eustachian Tube
- **Inner Ear**
  - Cochlea
  - Semicircular Canals

The Outer, Middle, and Inner Ear

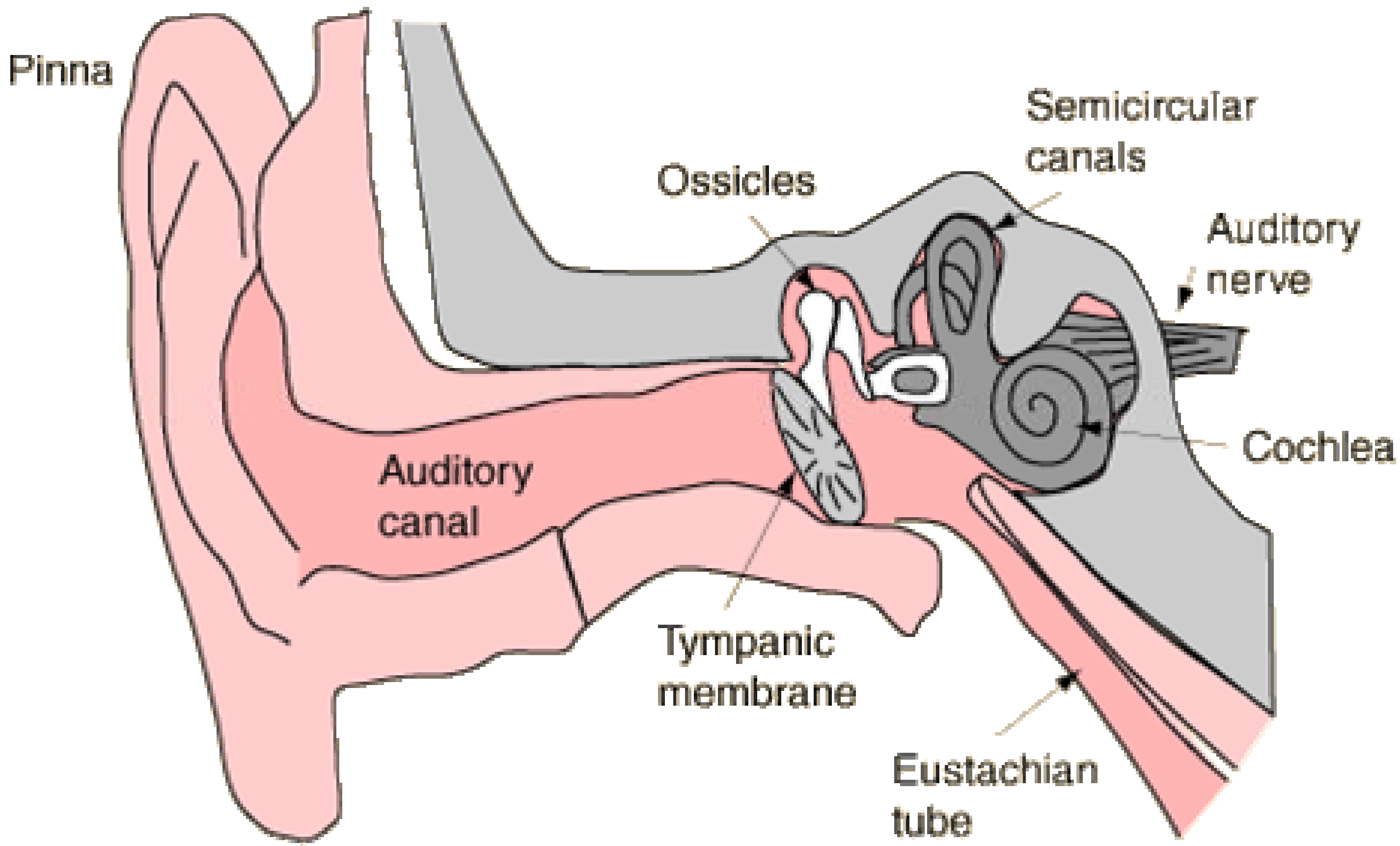


Image from http://hyperphysics.phy-astr.gsu.edu/hbase/hframe.html

T-61.184

The Outer Ear

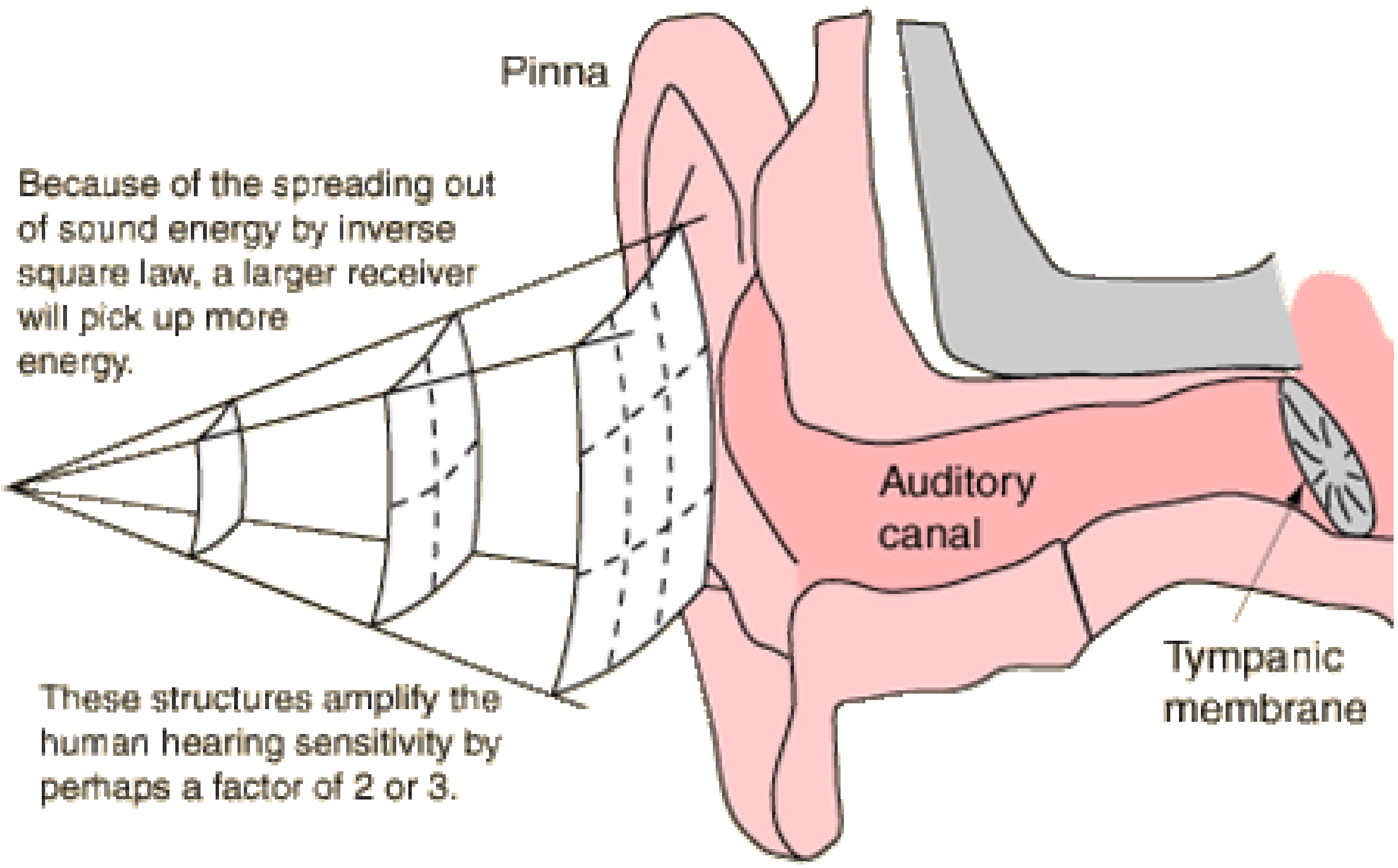
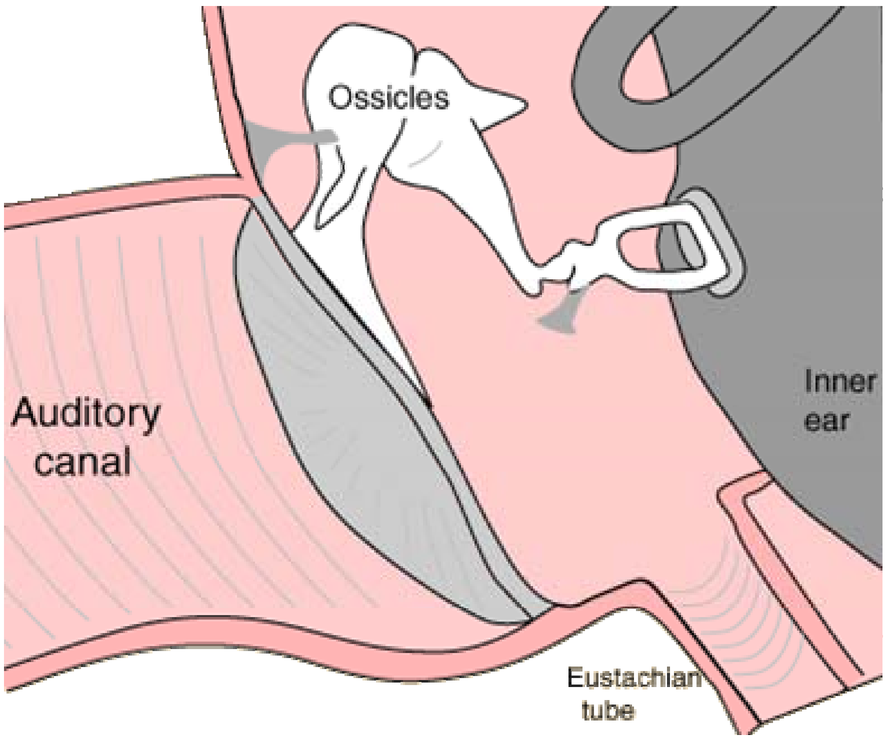


Image from http://hyperphysics.phy-astr.gsu.edu/hbase/hframe.html

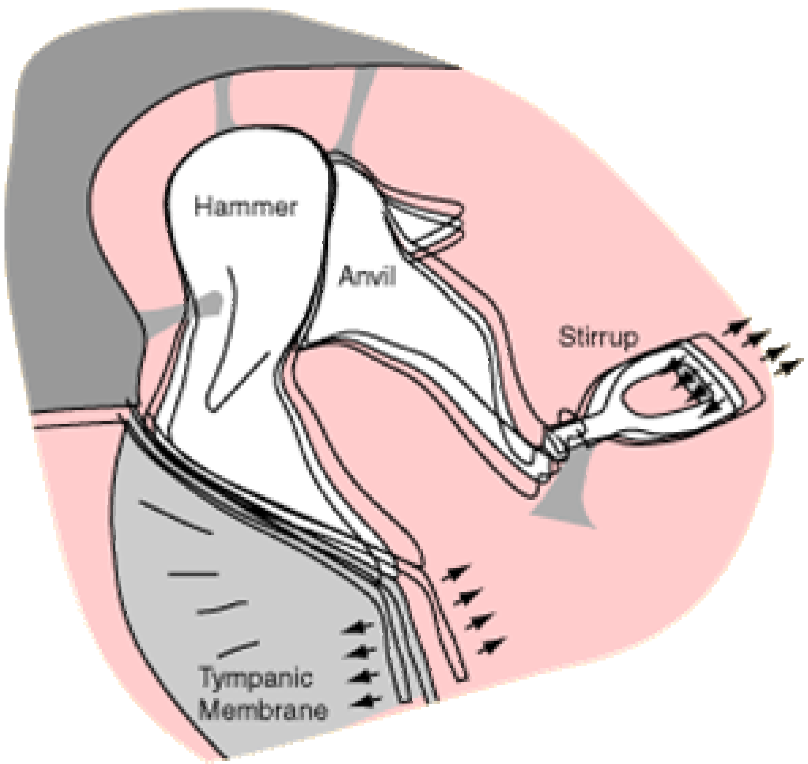
The Tympanic Membrane (Eardrum)



- Receives vibrations from auditory canal
- Transmits vibrations to Ossicles → then to Oval Widow (inner ear)
- Acts to amplify signal (eardrum is 15x larger in area than oval window)

Image from http://hyperphysics.phy-astr.gsu.edu/hbase/hframe.html

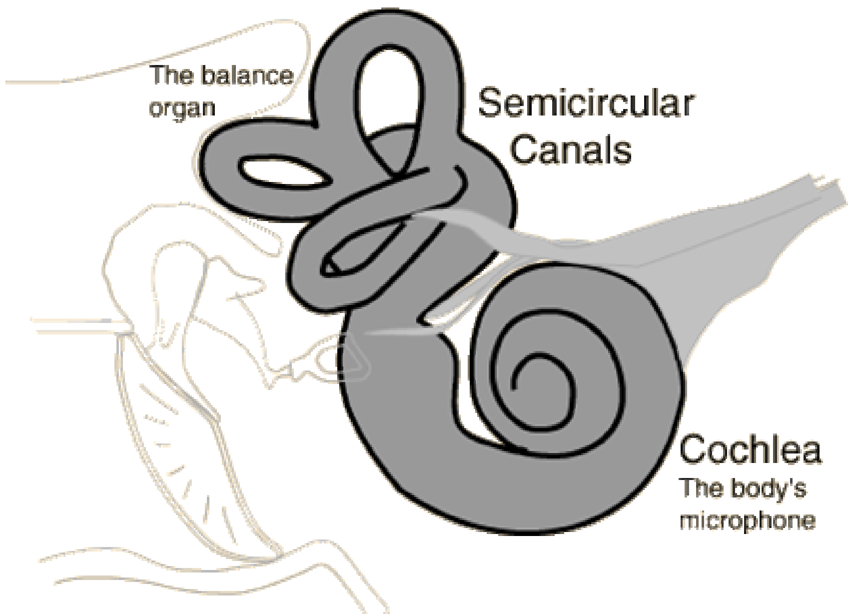
The Middle Ear



- **Ossicles:**
  - 3 bones: Malleus, Incus, Stapes
- **Amplifies signal by about a factor of 3**
- **Sends vibrations to the oval window (inner ear)**

Image from http://hyperphysics.phy-astr.gsu.edu/hbase/hframe.html

The Inner Ear



- **Semicircular Canals**
  - organs of balance
  - measure motion / acceleration
- **Cochlea**
  - (Cochlea = ‘Snail’ in Latin)
  - Acts as frequency analyzer
  - 2 ¾ turns
  - ~ 3.2 cm length

Image from http://hyperphysics.phy-astr.gsu.edu/hbase/hframe.html

Cochlea

- **Contains 3 fluid filled parts:**
  - ❑ (2) canals for transmitting pressure waves
    - ❑ Tympanic Canal
    - ❑ Vestibular Canal
  - ❑ (1) Organ of Corti
    - ❑ Senses Pressure changes
    - ❑ Perception of Pitch
    - ❑ Perception of Loudness

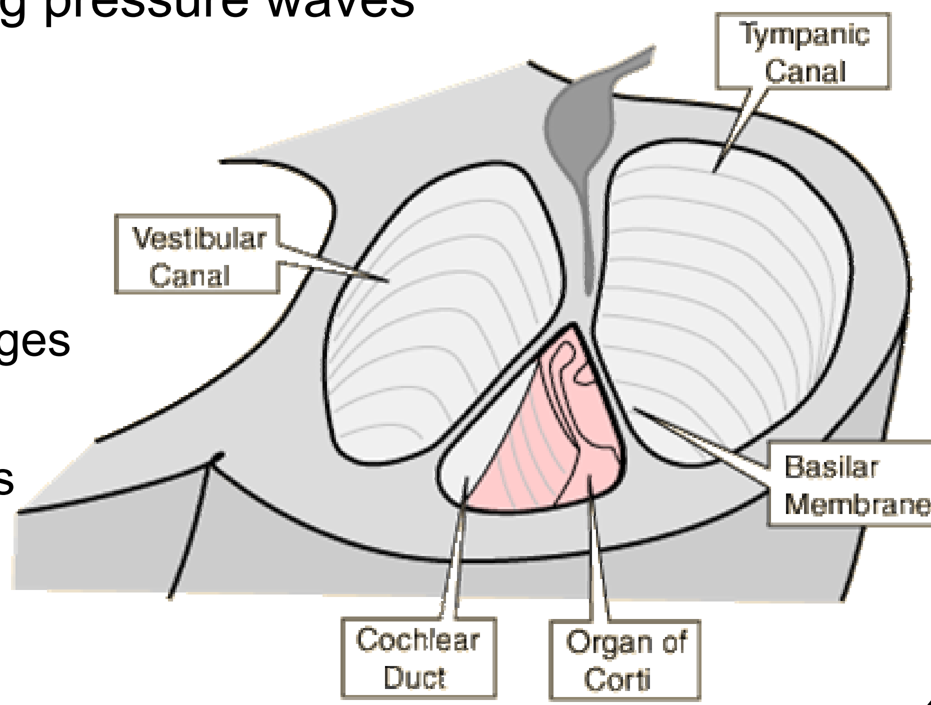
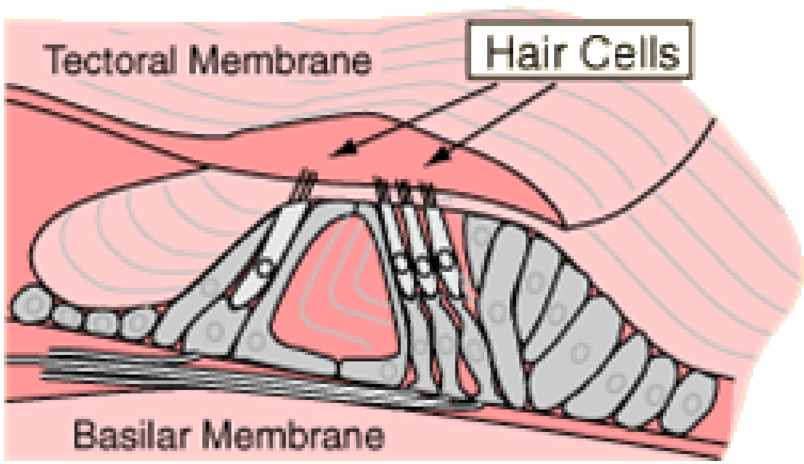


Image from http://hyperphysics.phy-astr.gsu.edu/hbase/hframe.html

The Organ of Corti

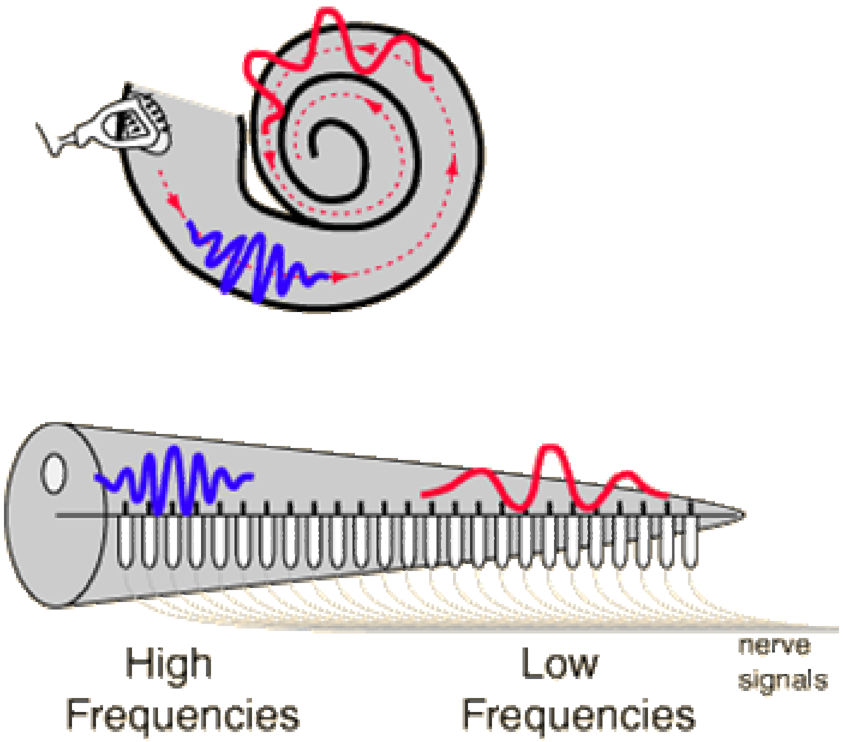


- Contains 4 rows of hair cells (~ 30,000 hair cells)
- Hair cells move in response to pressure waves in the vestibular and tympanic canals
- Hair cells convert motion into electrical signals
- Hair cells are tuned to different frequencies

Image from http://hyperphysics.phy-astr.gsu.edu/hbase/hframe.html

Place Theory:

Frequency Selectivity along the Basilar Membrane



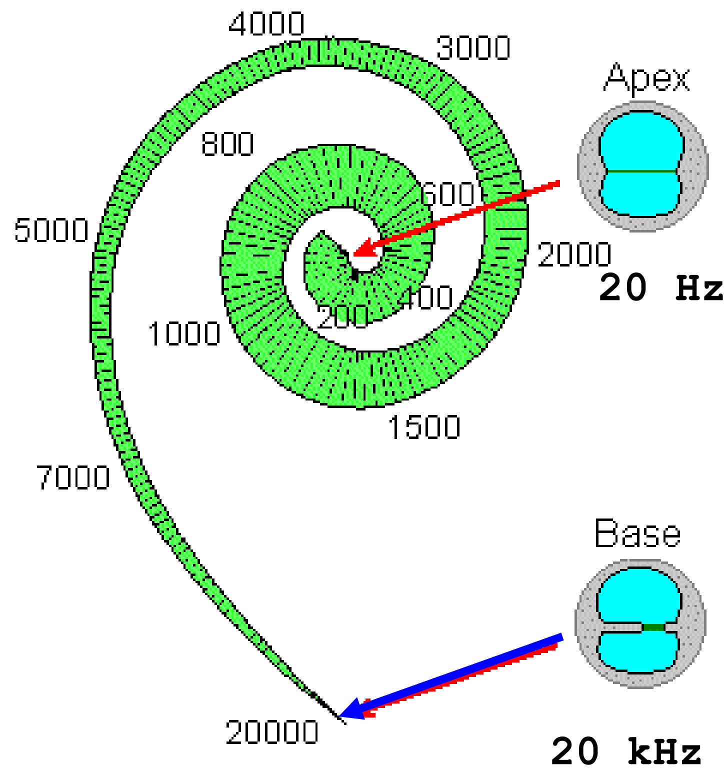
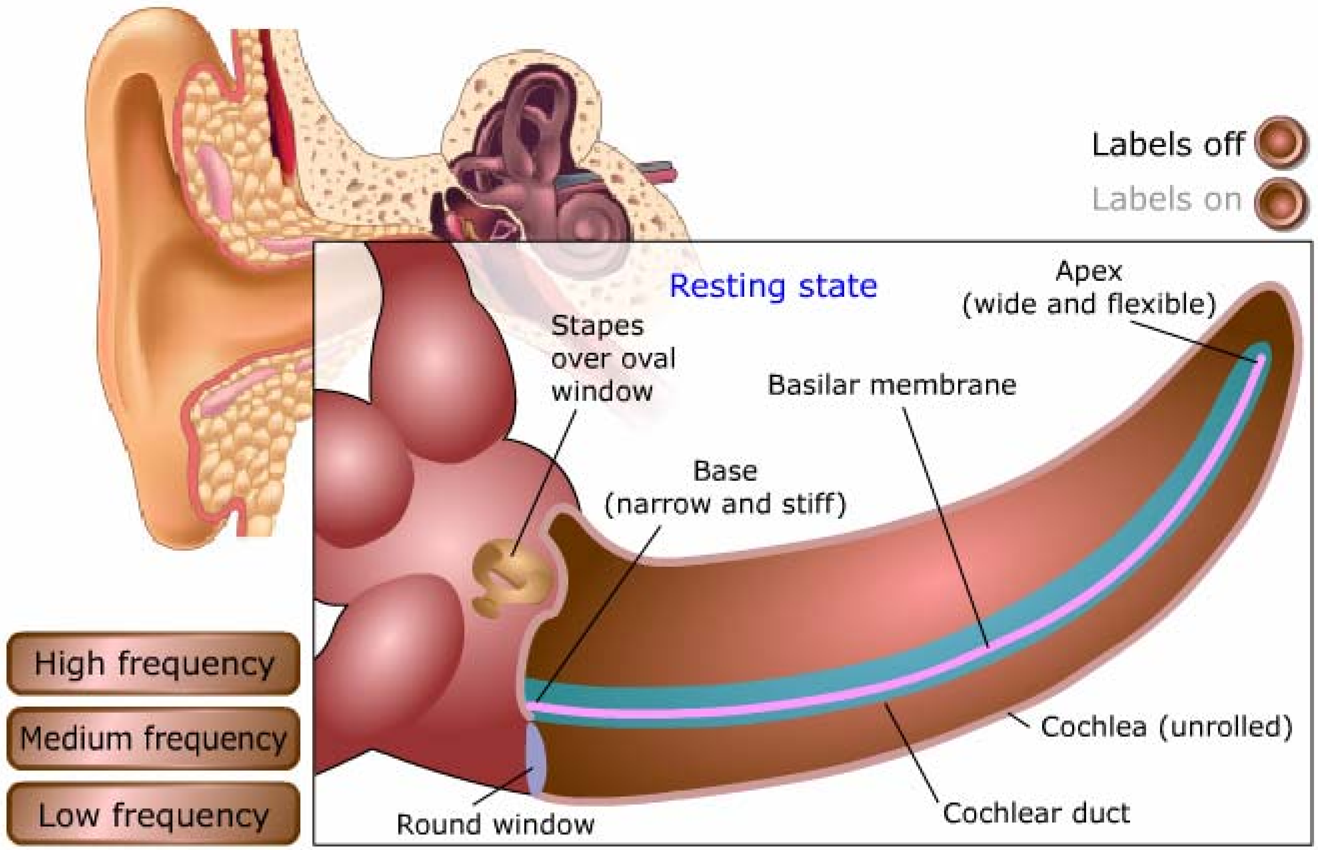


Image from http://hyperphysics.phy-astr.gsu.edu/hbase/hframe.html
http://www.iurc.montp.inserm.fr/cric/audition/english/ear/fear.htm

Graphical Example of Place Theory



From http://www.blackwellscience.com/matthews/

Summary

- **Outer Ear**
  - ❑ Sound waves travel down auditory canal to eardrum
- **Middle Ear**
  - ❑ Sound waves cause eardrum to vibrate
  - ❑ Ossicles (Malleus, Incus, Stapes) bones amplify and transmit sound waves to the Inner Ear (cochlea)
- **Inner Ear**
  - ❑ Cochlea acts like a spectrum analyzer
  - ❑ Converts sound waves to electrical impulses
  - ❑ Electrical Impulses travel down auditory nerve to the brain

Interesting Aspects of Perception

- **Audible sound range is from 20Hz to 20kHz**
- **Ear is not equally sensitive to all frequencies**
- **Perceived loudness is a function of both the frequency and the amplitude of the sound wave**

Intensity vs. Loudness

- **<u>Intensity</u>:** ***Physically measurable*** **quantity**
  - Sound power per unit area
  - Computed relative to the threshold of hearing:

$$I_0 = 10^{-12}\, watts / m^2$$

  - Measured on the decibel scale:

$$I(\mathrm{dB}) = \log_{10}\left[\frac{I}{I_0}\right]$$

source power

$$I = \frac{P}{4\pi r^2}$$

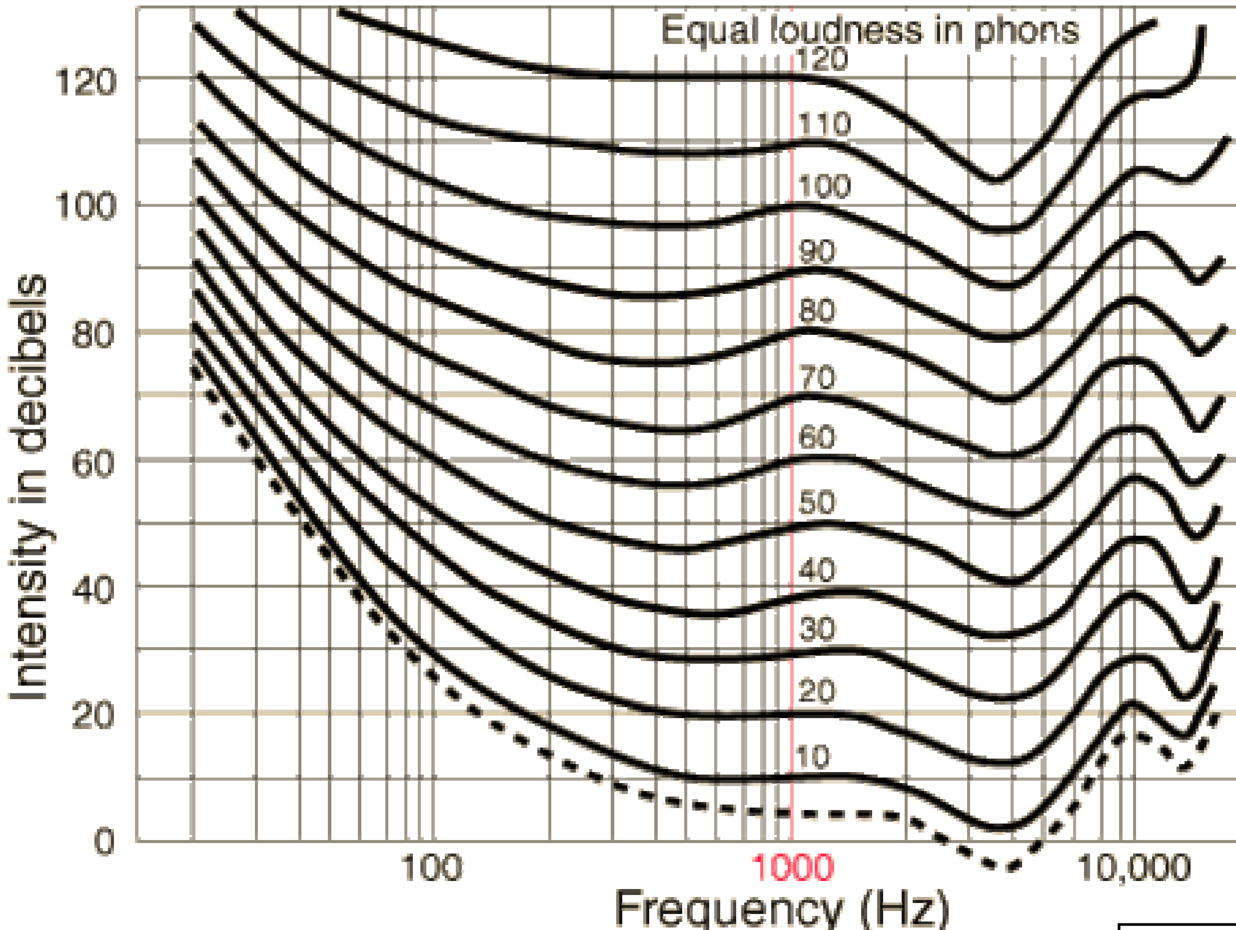
sphere area

- **<u>Loudness</u>:** ***Perceived*** **quantity**
  - Related to intensity
  - Ear's sensitivity varies with frequency

Loudness of Pure Tones

- **Contours of "equal loudness" can be estimated**
- **Labeled unit is the *phon* which is determined by the sound-pressure-level (SPL) in dB at 1kHz**
- **Ear relatively insensitive to low-frequency sounds of moderate to low-intensity**
- **Maximum sensitivity of the ear is at around 4kHz. There is a secondary local maximum near 13kHz due to the first two resonances of the ear canal**

Equal Loudness Curves



T-61.184

Loudness for Complex Tones

- **The total loudness of two pure tones, each having the same SPL, will be judged equal for frequency separations within a critical bandwidth. Once the frequency separation *exceeds* the critical bandwidth, however, the total loudness begins to increase.**
- **Broadband sounds will generally sound louder than narrow band (less than a critical bandwidth) sounds.**

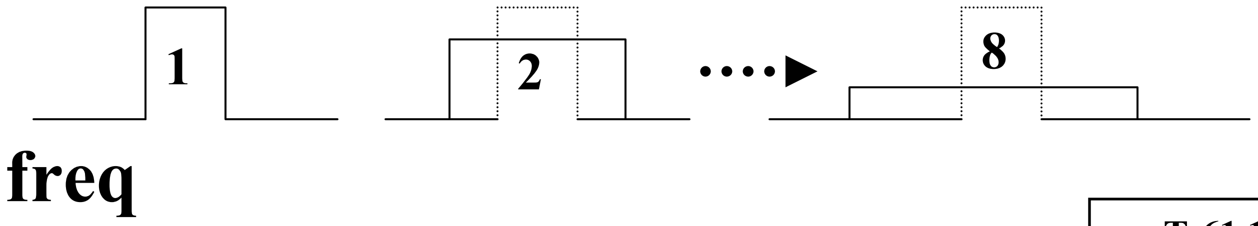
Critical Bands

- **Cochlea converts pressure waves to neural firings:**
  - Vibrations induce traveling waves down the basilar membrane
  - Traveling waves induce peak responses at frequency-specific locations on the basilar membrane
- **Frequency perceived within “critical bands”**
  - Act like band-pass filters
  - Defines “frequency resolution” of the auditory system
  - About 24 critical bands along basilar membrane.
  - Each critical band is about 1.3 mm long and embraces about 1300 neurons.

Measurement of Critical Bands

- **Two methods for measuring critical bands**
  - Loudness Method and Masking Method

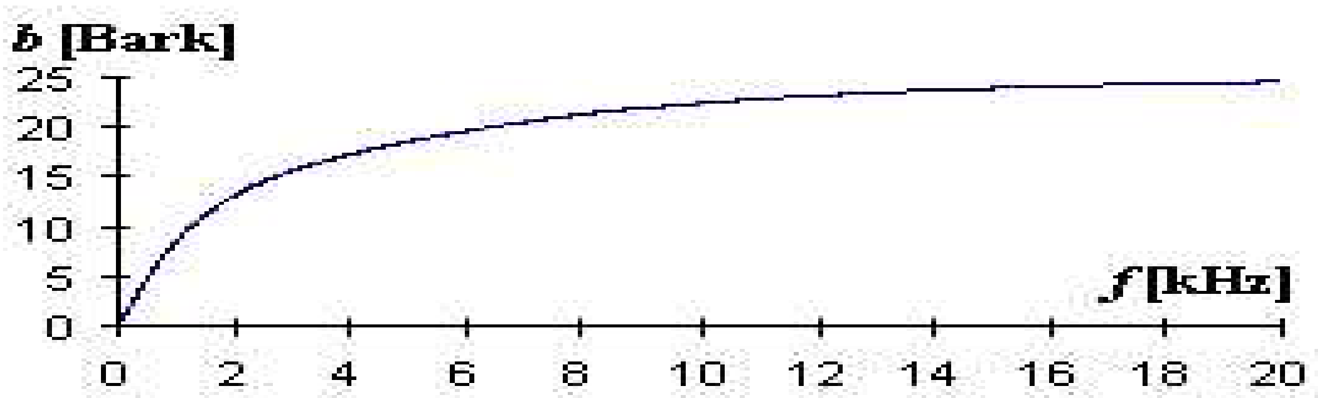
- **Loudness Method**
  - Bandwidth of a noise-burst is increased
  - Amplitude decreased to keep power constant
  - When bandwidth increases beyond critical band, subjective loudness increases (since the signal covers > 1 critical band)



Bark Frequency Scale

- A frequency scale on which equal distances correspond with perceptually equal distances.
- 1 bark = width of 1 critical band
- Above about 500 Hz this scale is more or less equal to a logarithmic frequency axis.
- Below 500 Hz the Bark scale becomes more and more linear.

Bark Scale



approx:

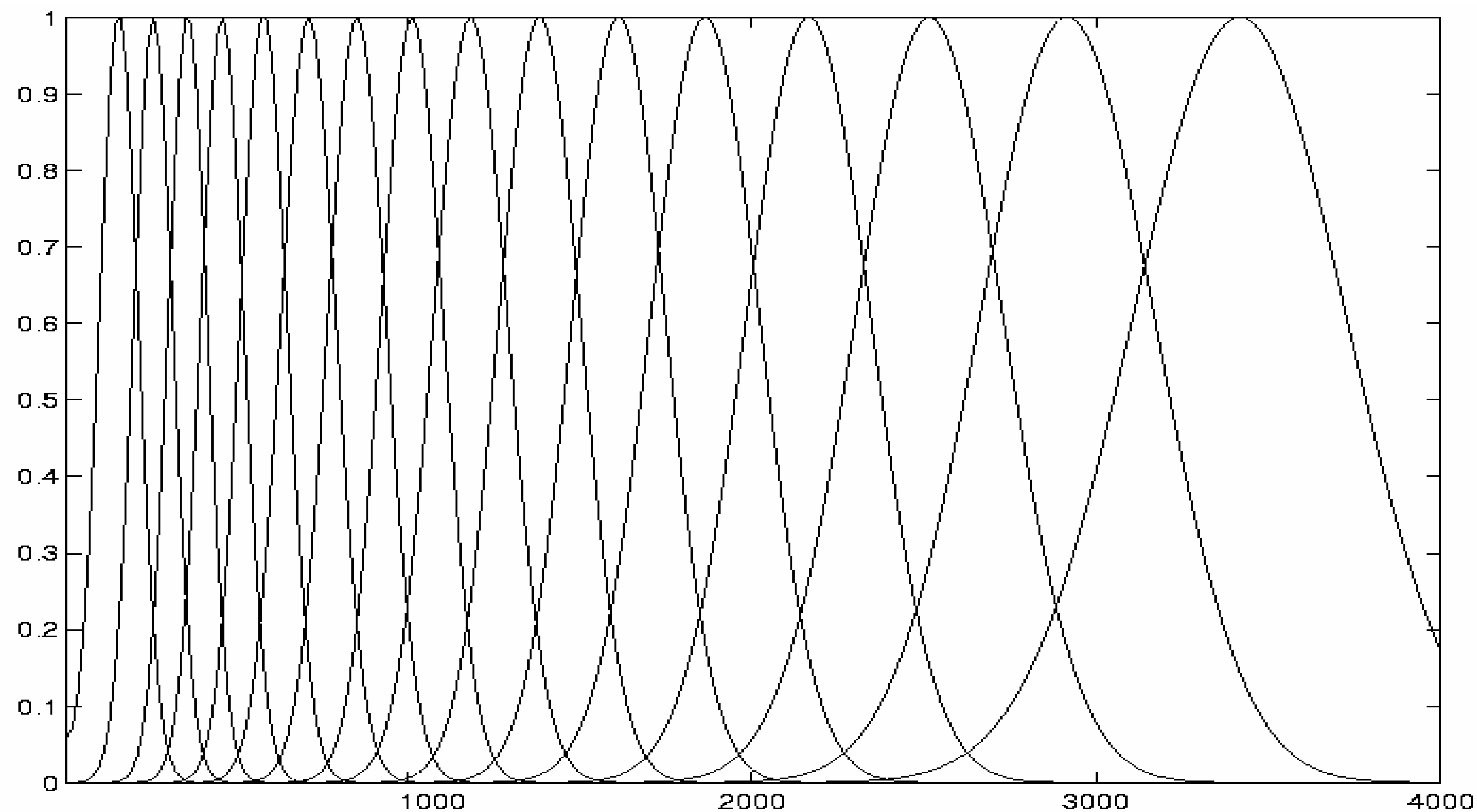
$$Bark(f) = \begin{cases} .01f, & 0 \le f < 500 \\ .007f + 1.5, & 500 \le f < 1220 \\ 6\ln(f) - 32.6, & 1220 \le f \end{cases}$$

or,

$$Bark(f) = \left\{ 13\text{atan}\left(\frac{0.76\text{f}}{1000}\right) + 3.5\text{atan}\left(\frac{\text{f}^2}{(7500)^2}\right) \right.$$

T-61.184

Bark Scale Bank of Filters



Mel Scale

- **Linear below 1 kHz and logarithmic above 1 kHz**
- **Based on perception experiments with tones:**
  - Divide frequency ranges into 4 perceptually equal intervals

    –or--
  - Adjust frequency of tone to be ½ as high as a reference tone
- **Approximation,**

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

T-61.184

Masking

- **Some sounds will "mask" or hide other sounds**
  - Depends on the relative frequencies and loudnesses of the two sounds.
- **Pure tones close together in frequency mask each other more than tones widely separated in frequency.**
- **A pure tone masks tones of higher frequency more effectively than tones of lower frequency.**
- **The greater the intensity of the masking tone, the broader the range of frequencies it can mask.**

Loudness and Duration

- **Loudness grows with duration, up to about 0.2 seconds.**
- **Muscles attached to the eardrum and ossicles provide protection from impulsive sounds (e.g., explosions, gunshots),**
  - ❑ Up to about 20 dB of protection is provided when exposed to sounds in excess of 85 dB
  - ❑ Reflex begins 30 to 40 ms after the sound overload and does not reach full protection for another 150 ms.

Speech Coding vs. Recognition

- Many advances have been made over the last 20 years in the area of speech coding (e.g., CELP, Mpeg-3, etc.)
- Coding techniques focus on modeling aspects of perception (e.g., masking, inaudibility of sounds, etc) to maximally model and compress speech
- Those techniques by-in-large <u>have not</u> been widely incorporated into the feature extraction stage of speech recognition systems. Why do you think this is so?

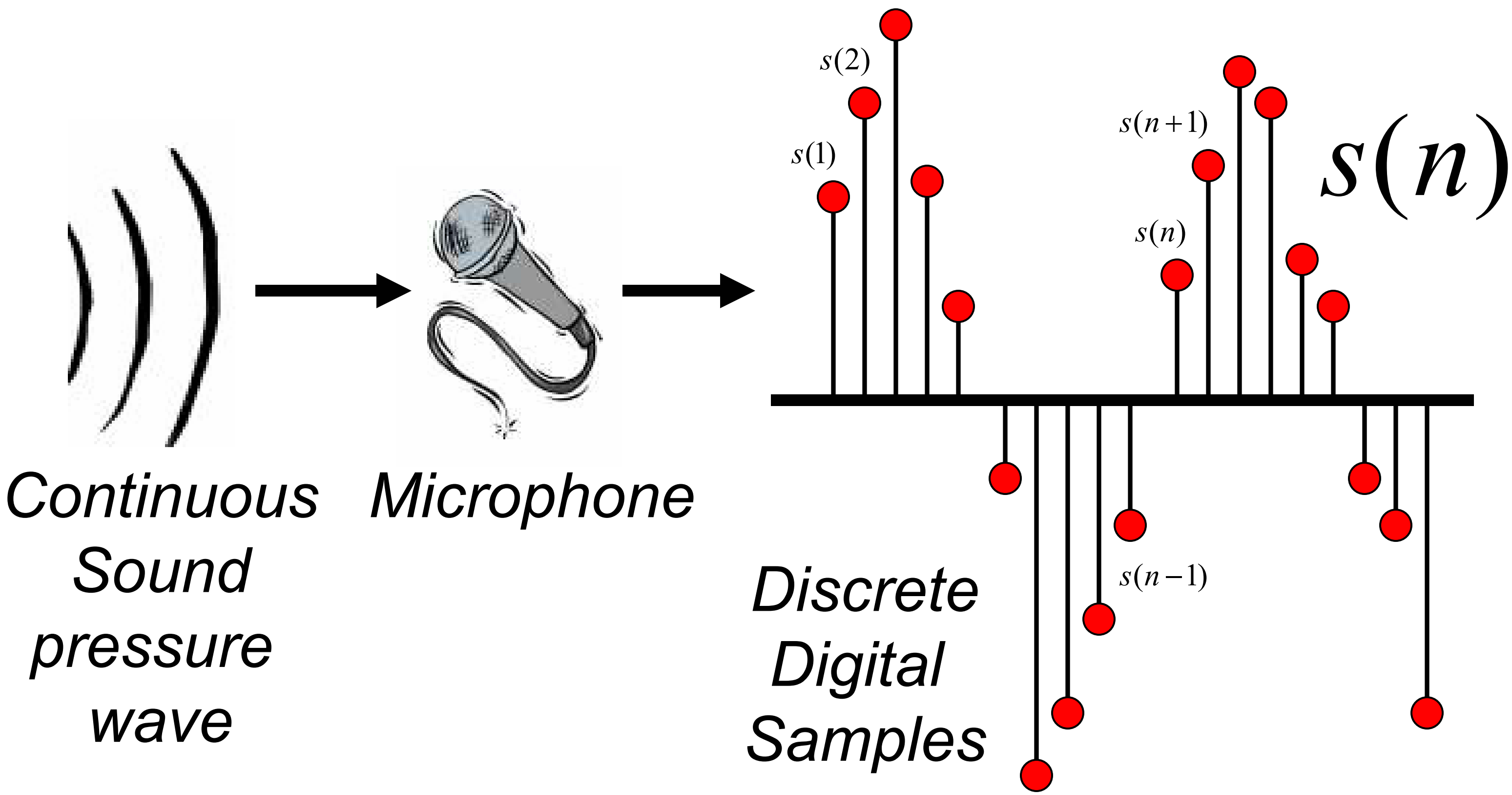
Feature Extraction for Speech Recognition

- **Frame-Based Signal Processing**
- **Linear Prediction Analysis**
- **Cepstral Representations**
  - Linear Prediction Cepstral Coefficients (LPCC)
  - Mel-Frequency Cepstral Coefficients (MFCC)
  - Perceptual Linear Prediction (PLP)

Goals of Feature Extraction

- Compactness
- Discrimination Power
- Low Computation Complexity
- Reliable
- Robust

Discrete Representation of Speech



Digital Representation of Speech

- **Sampling Rates**
  - 16,000 Hz (samples/second) for microphone speech
  - 8,000 Hz (samples/second) for telephone speech
- **Storage formats:**
  - Pulse Code Modulation (PCM)
    - 16-bit (2 bytes) per sample
    - +/- 32768 in value
    - Stored as “short” integers
  - Mu-Law and A-Law Compression
  - NIST Sphere “wav” files
  - Microsoft “wav” files

Practical things to Remember

- **Byte swapping is important**
  - Little-endian vs. Big-endian
- **Some audio formats have headers**
  - Sometimes we say “raw” audio to mean “no header”
  - Headers contain meta-information such as recording conditions and sampling rates and can be variable sized
  - Example formats: NIST Sphere, Microsoft wav, etc
- **Tip: Most Linux systems come with a nice tool called “sox” which can be used to convert signals from many formats into PCM bit-streams.   For a 16kHz Microsoft wav file:**

```
sox audiofile.wav -w -s -r 16000 audiofile.raw
```

Signal Pre-emphasis

- The source signal for voiced speech has an effective roll-off of -6dB/octave. Many speech analysis methods (e.g., linear prediction) work best when the source is spectrally flattened.
- Apply first order high-pass filter,

$$H(z) = 1 - az^{-1}, \qquad 0.9 \le a \le 1.0$$

- Implemented in time-domain as,

$$\tilde{s}(n) = s(n) - as(n-1)$$

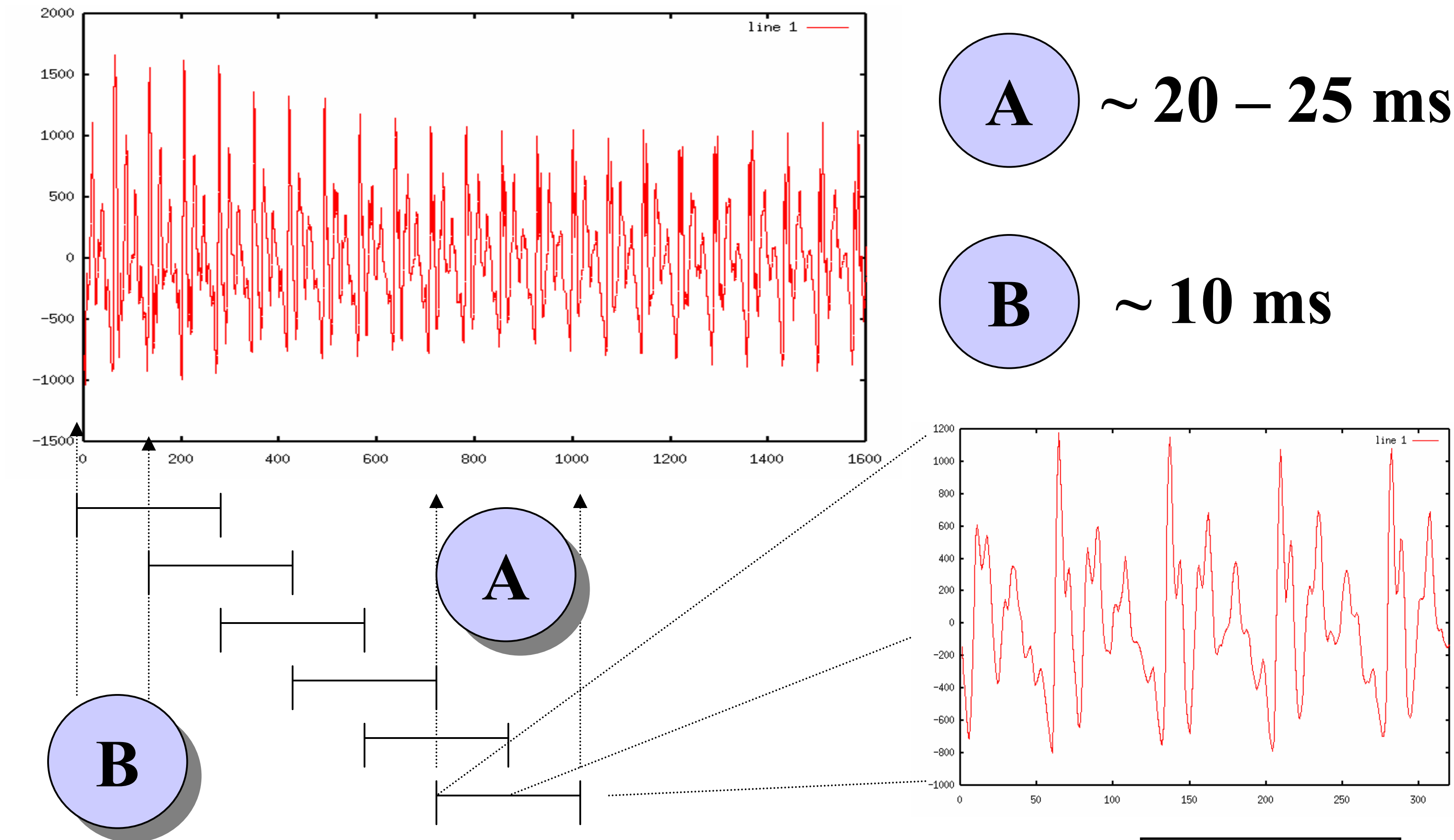
- "a" typically 0.95 to 0.97

Frame Blocking

- **Process the speech signal in small chunks over which the signal is assumed to have stationary spectral characteristics**
- **Typical analysis window is 25 msec**
  - 400 samples for 16kHz audio
- **Typical frame-rate is 10 msec**
  - Analysis pushes forward by 160 samples for 16kHz audio
- **Frames generally overlap by 50% in time**
  - Results in 100 “frames” of audio per second

T-61.184

Illustration of Frame Blocking

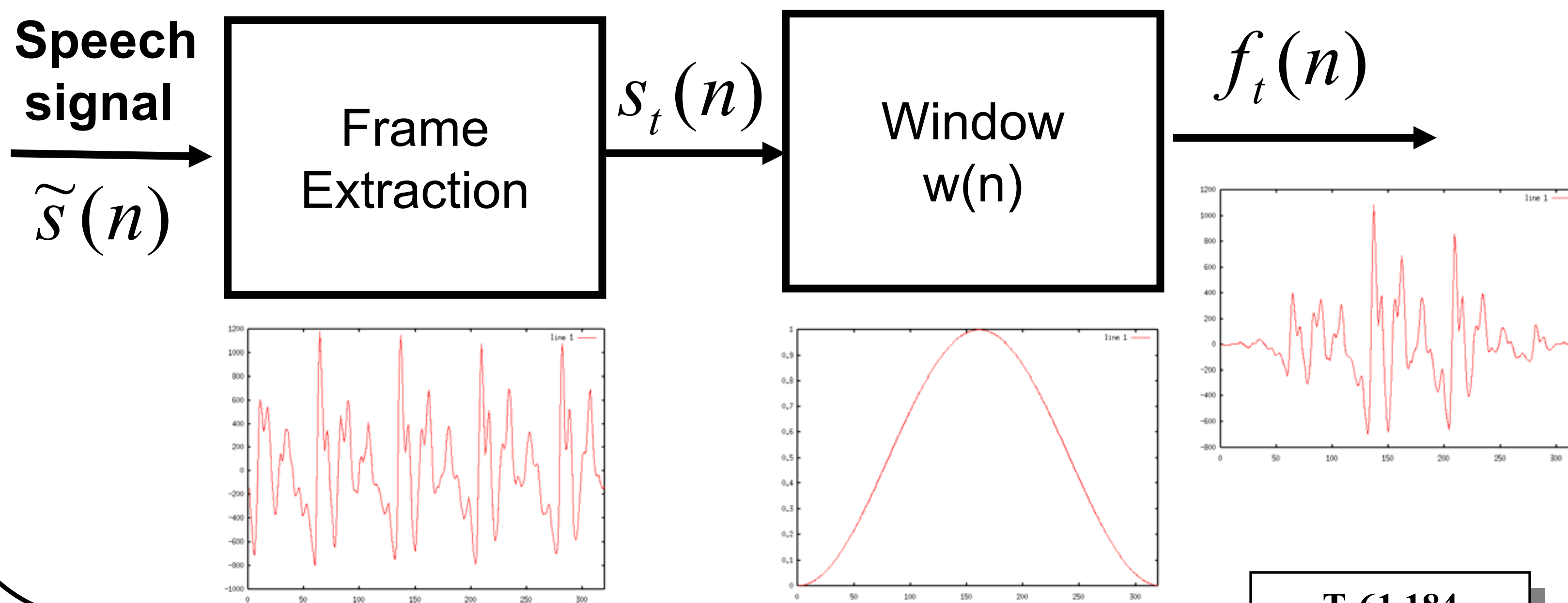


T-61.184

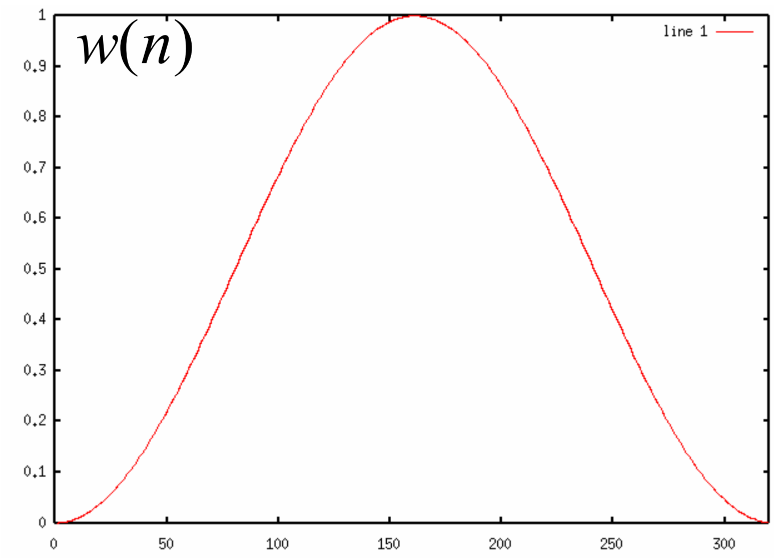
Frame Windowing

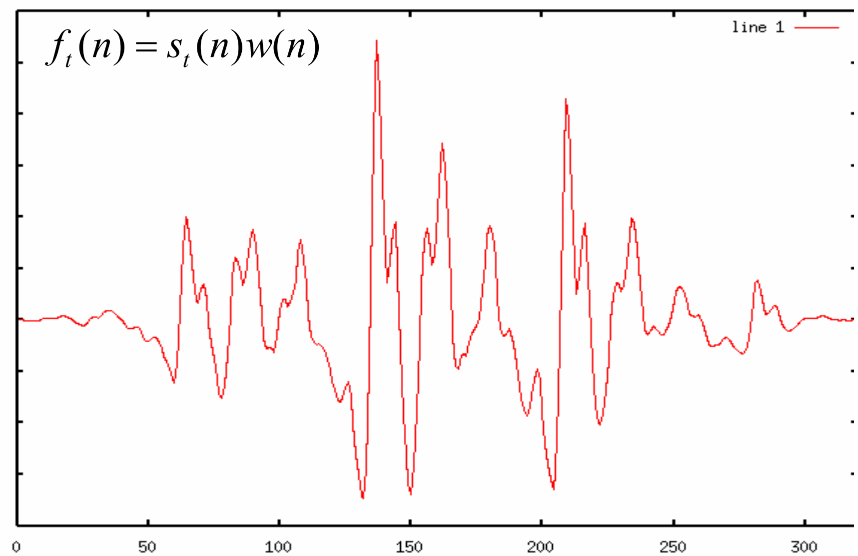
- **Each frame is multiplied by a smooth window function to minimize spectral discontinuities are the begin/end of each frame,**

$$f_t(n) = s_t(n)w(n)$$



Example: Hanning Window





$$w(n) = \begin{cases} \frac{1}{2}\cos\left(1 - \left(\frac{2\pi \cdot n}{N}\right)\right), & \text{n} = 0,1,\ldots,\text{N}-1 \\ 0, & \text{n} \quad \text{otherwise} \end{cases}$$

T-61.184

Alternative Window Function

- **Can also use the Hamming window,**

$$w(n) = \begin{cases} 0.54 - 0.46\cos\left(\dfrac{2\pi(n-1)}{N-1}\right), & \mathrm{n} = 0,1,\ldots,\mathrm{N}-1 \\ 0, & \mathrm{n} \quad \text{otherwise} \end{cases}$$

- **Default window used by the Cambridge HTK system**

Frame-based Processing Example: Speech Detection

- **Accurate detection of speech in the presence of background noise is important to limit the amount of processing that is needed for recognition**

- **Endpoint-Detection algorithms must take into account difficult situations such as,**
  - Utterances that contain low-energy events at beginning/end (e.g., weak fricatives)
  - Utterances ending in unvoiced stops (e.g., ‘p’, ‘t’, ‘k’)
  - Utterances ending in nasals (e.g., ‘m’,’n’).
  - Breath noises at the end of an utterance

End-Point Detection

- **End-Point Detection Algorithms mainly assume the entire utterance is known. Must search for begin and end of speech**

- **Rabiner and Sambur, "An Algorithm for Determining the Endpoints of Isolated Utterances". The Bell System Technical Journal, Vol. 54, No. 2, February 1975, pp. 297-315**

- **Proposed end-point algorithm based on,**
  - ITU - Upper energy threshold.
  - ITL - Lower energy threshold.
  - IZCT - Zero crossings rate threshold.

Energy and Zero-Crossing Rate

- **Log-Frame Energy**
  - log of the square sum of the signal samples

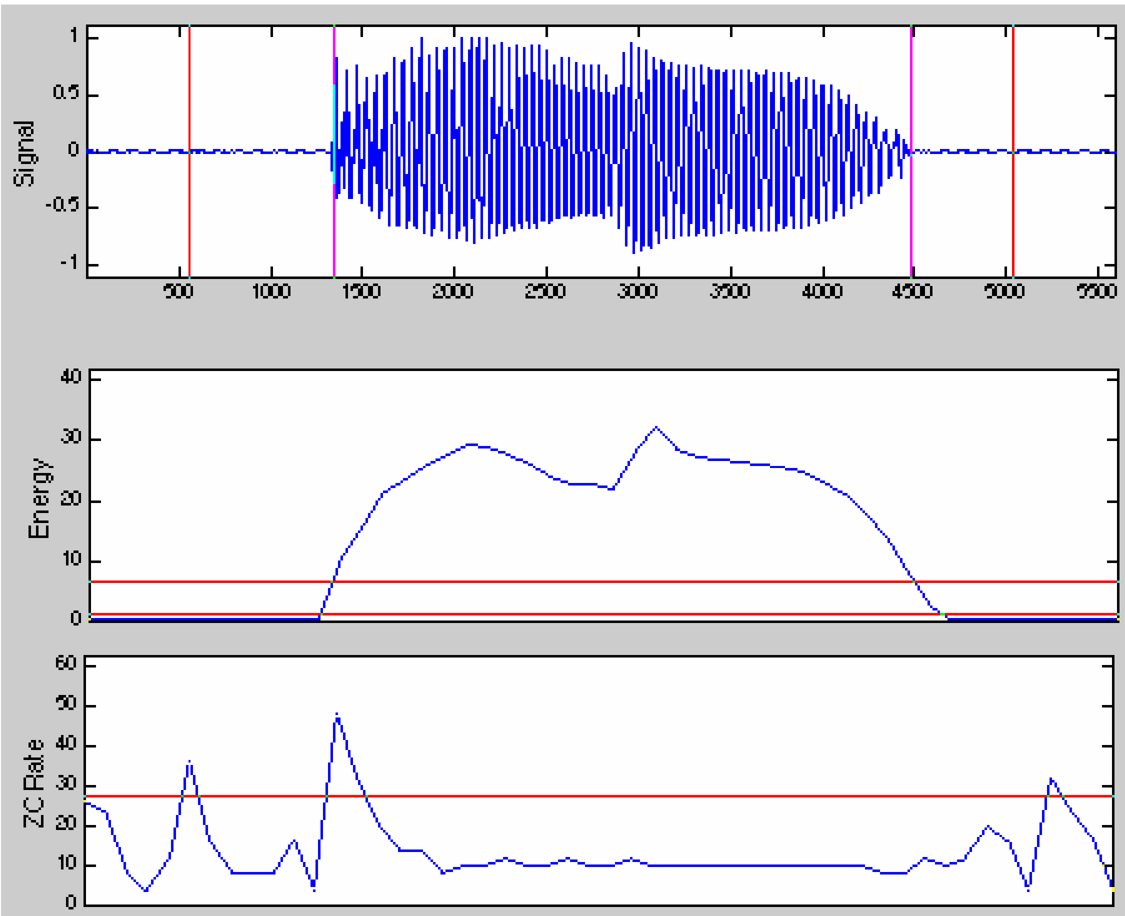
- **Zero-Crossing Rate**
  - Frequency at which the signal cross the 0 axis

$$ZCR = 0.5\sum_{i=1}^{N}\left[\text{sign}(s(i)) - \text{sign}(s(i-1))\right]$$

Idea of the Rabiner / Sambur Algorithm

- **Begin-Point:**
  - Search for the first time the signal exceeds the upper energy threshold (ITU).
  - Step backwards from that point until the energy drops below the lower energy threshold (ITL).
  - Consider previous 250 msec of zero-crossing rate. If ZCR exceeds IZCT threshold 3 or more times, set begin point to the first occurrence that threshold is exceeded
- **End-Point:**
  - Similar to begin-point algorithm but takes place in the reverse direction.

End-Point Detection Example



Signal

Signal energy with ITU and ITL Thresholds

Zero-Crossing Rate with IZCT Threshold

Figure from http://www.dcs.shef.ac.uk/~martin/MAD/epd/epd.htm

Linear Prediction (LP) Model

- **Samples from a windowed frame of speech can be predicted as a linear combination of P previous samples and error u(n):**

$$s(n) = \sum_{k=1}^{P} a_k s(n-i) + G \cdot u(n)$$

- **u(n) is an excitation source and G is the gain of the excitation. The a_i terms are the LP coefficients and P is the order of the model.**

Linear Prediction (LP) Model

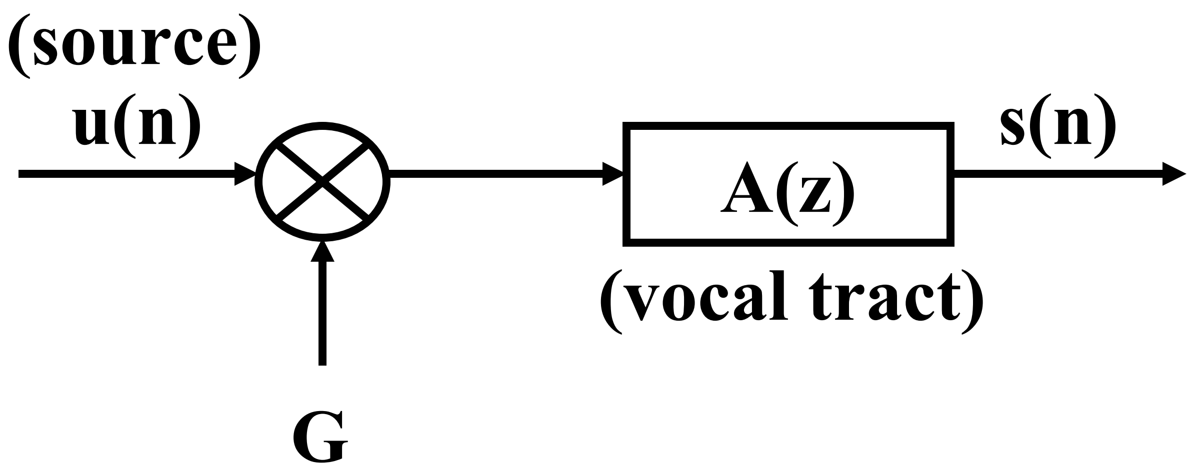
- **In the Z-domain,**

$$S(z) = \sum_{k=1}^{P} a_k z^{-1} S(z) + G \cdot U(z)$$

- **Results in a transfer function,**

$$H(z) = \frac{S(z)}{U(z)} = \frac{G}{1 - \sum_{k=1}^{P} a_k z^{-1}} = \frac{G}{A(z)}$$

Linear Prediction (LP) Model



- **u(n) assumed to be an impulse train for voiced speech and random white noise for unvoiced speech.**

Computing LP Parameters

- **Model:** $$s(n) = \sum_{i=1}^{P} a_i s(n-i) + G \cdot u(n)$$
- **Prediction:** $$\tilde{s}(n) = \sum_{k=1}^{P} a_k s(n-k)$$
- **Model Error:** $$e(n) = s(n) - \tilde{s}(n)$$
- **Minimize MSE:** $$E = \frac{1}{N} \sum e^2(n)$$

T-61.184

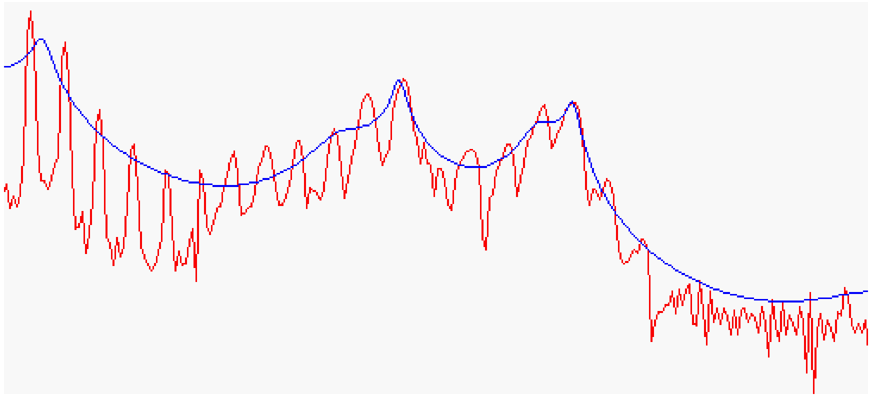
Computing LP Parameters

- **The model parameters are found by taking the partial derivative of the MSE with respect to the model parameters.**
- **Can be shown that the parameters can be solved quite efficiently by computing the autocorrelation coefficients from the speech frame and then applying what is known as the Levinson-Durbin recursion.**

LP Parameter Estimation

- **LP model provides a smooth estimate of spectral envelope.**

$$H\left(e^{j\omega}\right)=\frac{G}{A\left(e^{j\omega}\right)}$$



- **Typically model orders (P) are**
  - P between 8→12 for 8kHz audio,
  - P between 16→24 for 16kHz audio.

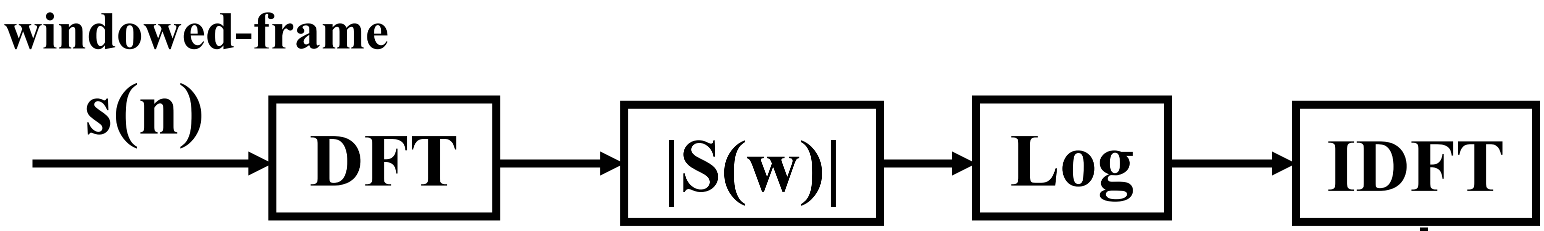
Cepstral Analysis of Speech

- **Want to separate the source (E) from the filter (H),**

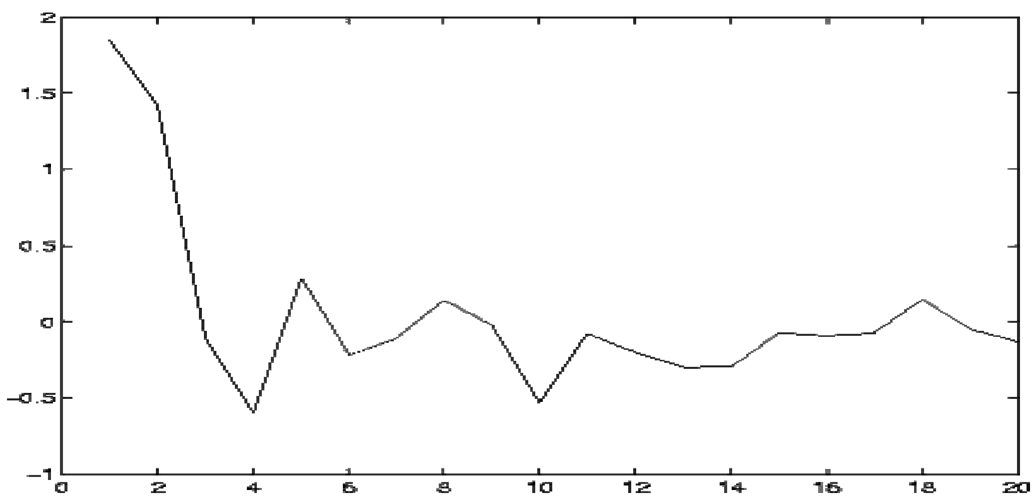
$$S(e^{j\omega}) = H(e^{j\omega})E(e^{j\omega})$$
$$\log\left\{\left|S(e^{j\omega})\right|\right\} = \log\left\{\left|H(e^{j\omega})\right|\right\} + \log\left\{\left|E(e^{j\omega})\right|\right\}$$

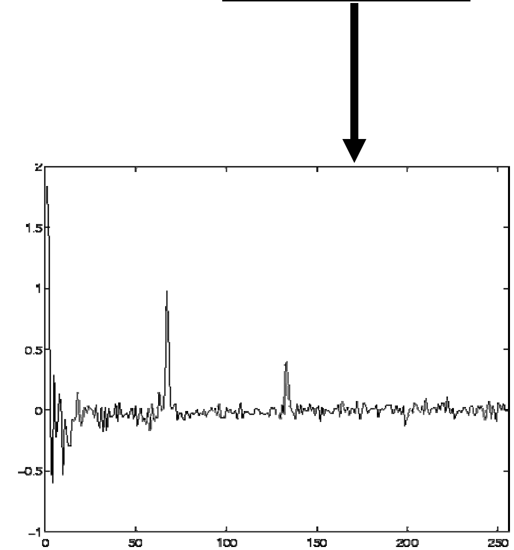
- **"E" roughly represents the excitation and "H" represents the contribution from the vocal tract.**
- **Slowly varying components of log-spectrum represented by low frequencies and fine detail by higher frequencies**
- **<u>Cepstral coefficients</u> are the coefficients derived from the Fourier transform of the log-magnitude spectrum**

Cepstral Analysis of Speech



- Keep first N coefficients to represent vocal tract (typically N=12 to 14)





cepstrum

LP Cepstral Coefficients (LPCC)

- **Simple recursion to convert LP parameters to cepstral parameters for speech recognition,**

$$c_0 = \ln \sigma^2$$

$$c_m = a_m + \sum_{k=1}^{m-1} \left(\frac{k}{m} \right) c_k a_{m-k} \qquad 1 \le m \le P$$

$$c_m = \sum_{k=1}^{m-1} \left(\frac{k}{m} \right) c_k a_{m-k} \qquad m > P$$

- **Typically 12-14 coefficients are computed**

Liftering

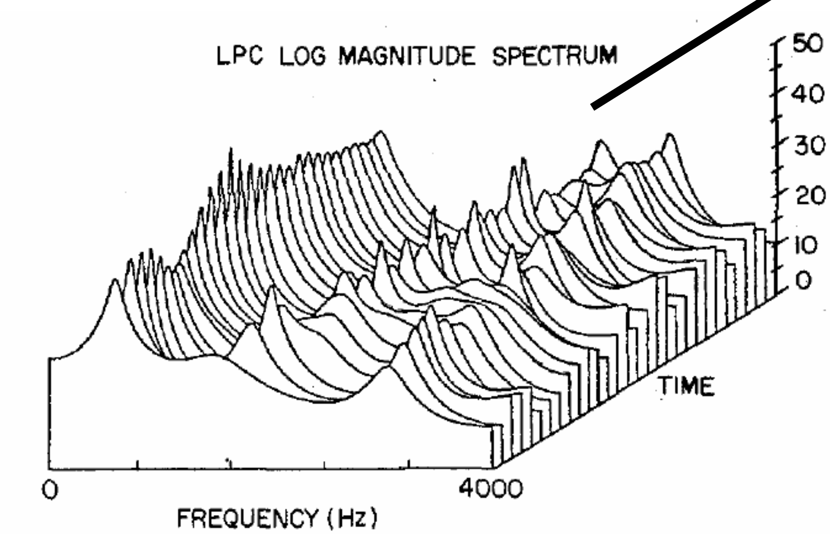
- **High-order cepstral coefficients can be numerically small.**
- **Common solution to lifter (scale) the coefficients,**

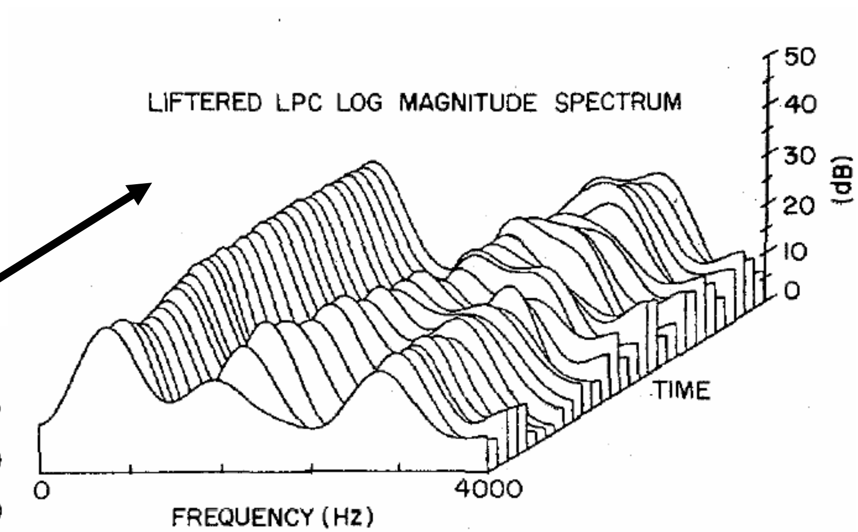
$$c'_m = \left(1 + \frac{L}{2}\sin\left(\frac{\pi m}{L}\right)\right)c_m$$

- **Default value for L is 22 for a 12th order cepstral vector (m=1..12) for the Cambridge HTK recognizer**

Impact of Liftering

- **Essentially reduces variability in LPC spectral measurements**





Biing-Hwang Juang; Rabiner, L.; Wilpon, J."On the use of bandpass liftering in speech recognition", IEEE Transactions on Acoustics, Speech, and Signal Processing, Vol.ASSP-35, No. 7, pp. 947-954, 1987

Additive Noise and LPCCs

- **Mansour and Juang (1989) studied LPCCs in additive white noise and found,**

  1. The means of cepstral parameters shift in noise
  2. The norm of Cepstral vectors is reduced in noise
     - Vectors with large norms are less effected by noise compared to vectors with smaller norms
     - Lower order coefficients are more affected compared to higher order coefficients
  3. The direction of the cepstral vector is less sensitive to noise compared to the vector norm.

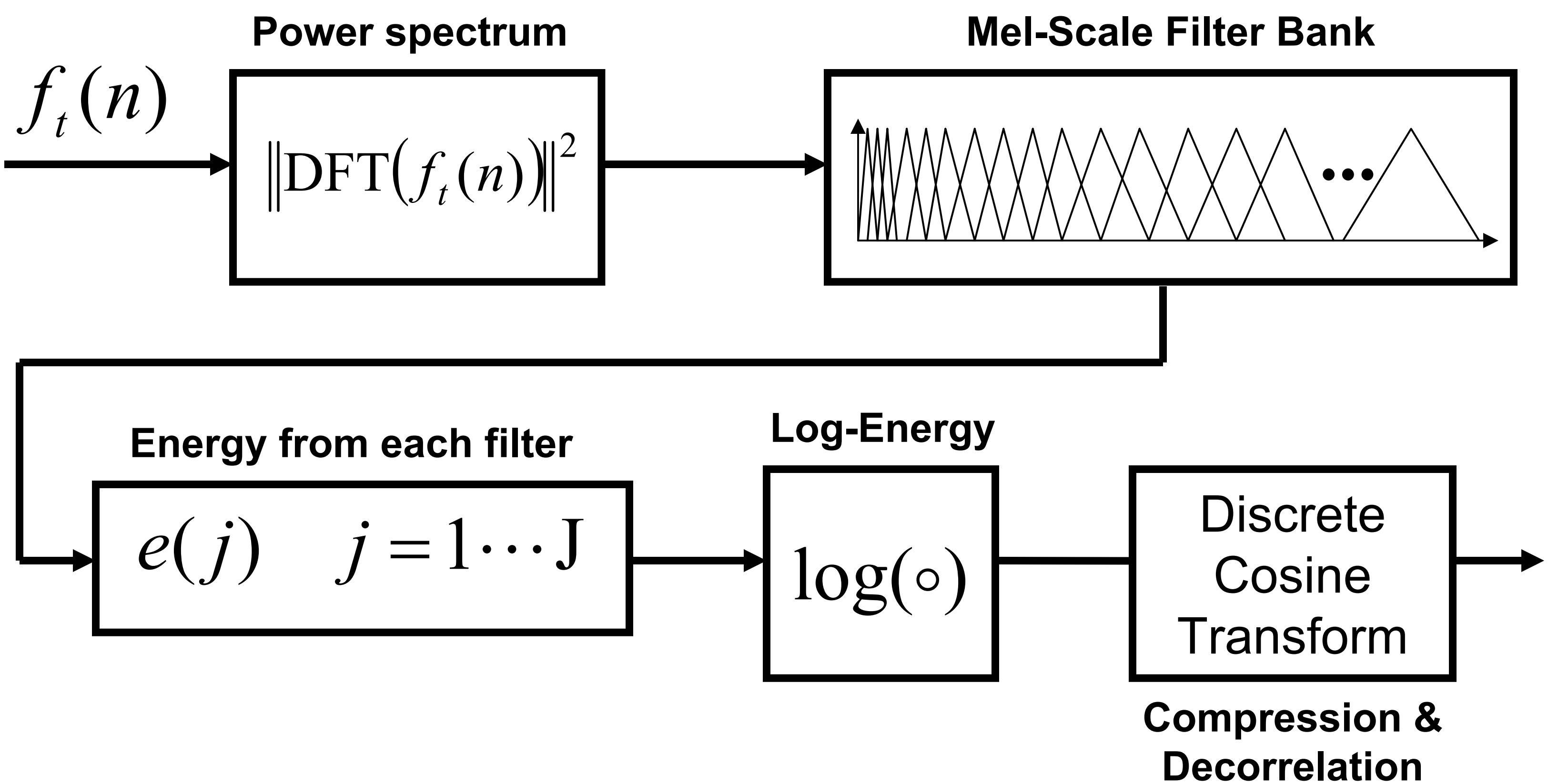
- **They proposed a projection measure for distance calculation based on this finding. Useful for earlier recognizers based on template matching.**

Mel-Frequency Cepstral Coefficients (MFCC)

- Davis & Mermelstein (1980)
- Computes signal energy from a bank of filters that are linearly spaced at frequencies below 1kHz and logarithmically spaced above 1kHz.
- Same and equal spacing of filters along Mel-Scale,

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

MFCC Block Diagram



Mel-Scale Filterbank Implementation

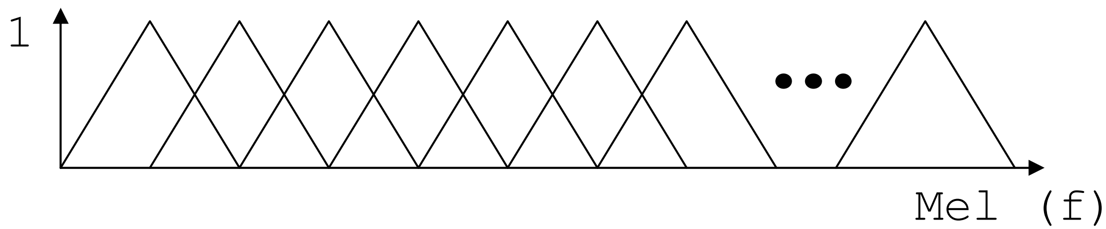
- (20-24) triangular shaped filters spaced evenly along the Mel Frequency Scale with 50% overlap
- Energy from each filter is computed (N = DFT size, P=# filters) at time t:

$$e[j][t] = \sum_{k=0}^{N-1} \underbrace{H_j[k]}_{\textit{Triangular Filter}} \cdot \underbrace{\left\| \tilde{S}_t[k] \right\|^2}_{\textit{Signal Power Spectrum}} \qquad \text{for } j = 1 \ldots P$$

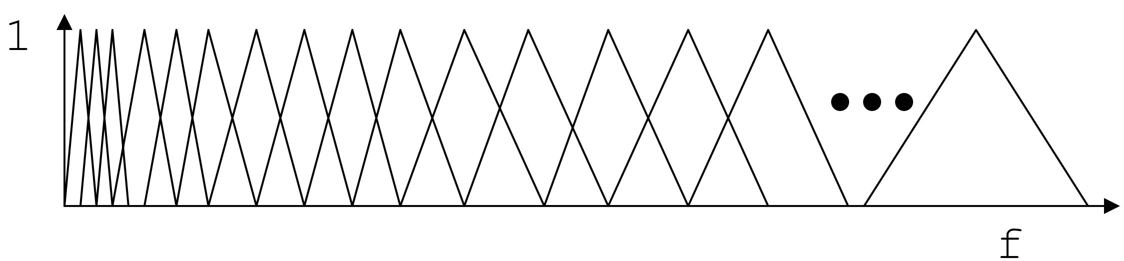
T-61.184

Mel-Scale Filterbank Implementation

- Equally spaced filters along the Mel-frequency scale with 50% overlap



- Analogous to non-uniformly spaced filters along linear frequency scale:



Final Steps of MFCC Calculation

- **Compute Log-Energies from each of P filters**
- **Apply Discrete Cosine Transform (DCT)**

$$MFCC[i][t] = \sqrt{\frac{2}{P}} \sum_{j=1}^{P} \left\{ \left(\log e[j][t]\right) \cdot \cos\left(\frac{\pi i}{P} (j - 0.5) \right) \right\}$$

- **DCT: (1) improves diagonal covariance assumption, (2) compresses features**
- **Typically 12-14 MFCC features are extracted (higher order MFCCs useful for speaker-ID)**

Why are MFCC's still so Popular?

- **Efficient (and relatively straight forward) to compute**
- **Incorporate a perceptual frequency scale**
- **Filter banks reduce the impact of excitation in the final feature sets**
- **DCT decorrelates the features**
  - Improves diagonal covariance assumption in HMM modeling that we will discuss soon

Perceptual Linear Prediction (PLP)

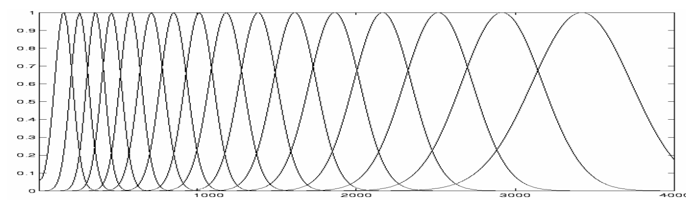
- **H. Hermansky. "Perceptual linear predictive (PLP) analysis of speech". *Journal of the Acoustical Society of America*, 87:1738-1752, 1990.**

- **Includes perceptual aspects into recognizer**
  - equal-loudness pre-emphasis
  - intensity-to-loudness conversion

- **More robust than linear prediction cepstral coefficients (LPCCs).**

(Rough) PLP Block-Diagram

$f_t(n)$ → **Power spectrum** $\left\| \mathrm{DFT}\left(f_t(n)\right) \right\|^2$ → **Bark-Scale Frequency Warping and Filter Bank Convolution**



$$\mathrm{Bark}(f) = 6\ln\left\{ \frac{f}{1200} + \left[\left(\frac{f}{1200}\right)^2 + 1 \right]^{0.5} \right\}$$

Equal Loudness Pre-emphasis → **Intensity-to-loudness** → **Linear Prediction Analysis** → **Compute LP Cepstral Coefficients** →

$$E(f) = \left(\frac{f^2}{f^2 + 1.6e5} \right)^2 \cdot \frac{f^2 + 1.44e6}{f^2 + 9.61e6}$$

$$\Phi(f) = \Psi(f)^{0.33}$$

PMVDR Cepstral Coefficients

- **Perceptual <u>M</u>inimum <u>V</u>ariance <u>D</u>istortionless <u>R</u>esponse (PMVDR) Cepstral Coefficients**
- **Based on MVDR spectral representation**
  - Improves modeling of upper envelope of speech signal
- **Shares some similarities to PLP**
- **Does not require the filter bank implementation of PLP, LPCC, or MFCC features**

MVDR Spectral Estimation

- Capon (1969)
- The signal power at a given frequency, ω_l, is estimated by designing an *M*th order FIR filter, $h_l(n)$, that minimizes its output power subject to the constraint that the response at the frequency of interest (ω_l) has unity gain
- This constraint is known as a *distortionless* constraint.

MVDR Spectral Estimation

- The Mth order MVDR spectrum of a frame of speech is obtained from the LP coefficients (a's) and LP prediction error (Pe),

$$S_{MV}^{(M)}(\omega) = \frac{1}{\sum_{k=-M}^{M} \mu_k e^{-j\omega k}}$$

$$\mu_k = \begin{cases} \frac{1}{P_e} \sum_{i=0}^{M-k} (M+1-k-2i) a_i a_{i+k}^*, & k = 0,\ldots,M \\ \mu_{-k}^* & k = \text{-}M,\ldots\text{-}1 \end{cases}$$

T-61.184

MVDR Spectral Estimation

- MVDR has been shown to provide improved tracking of the upper envelope of the signal spectrum (Murthi & Rao, 2000)
- Suitable for modeling voiced and unvoiced speech
- Provides smoother estimate of signal spectrum compared to LP. Makes it more robust to noise

MVDR Spectrum Example

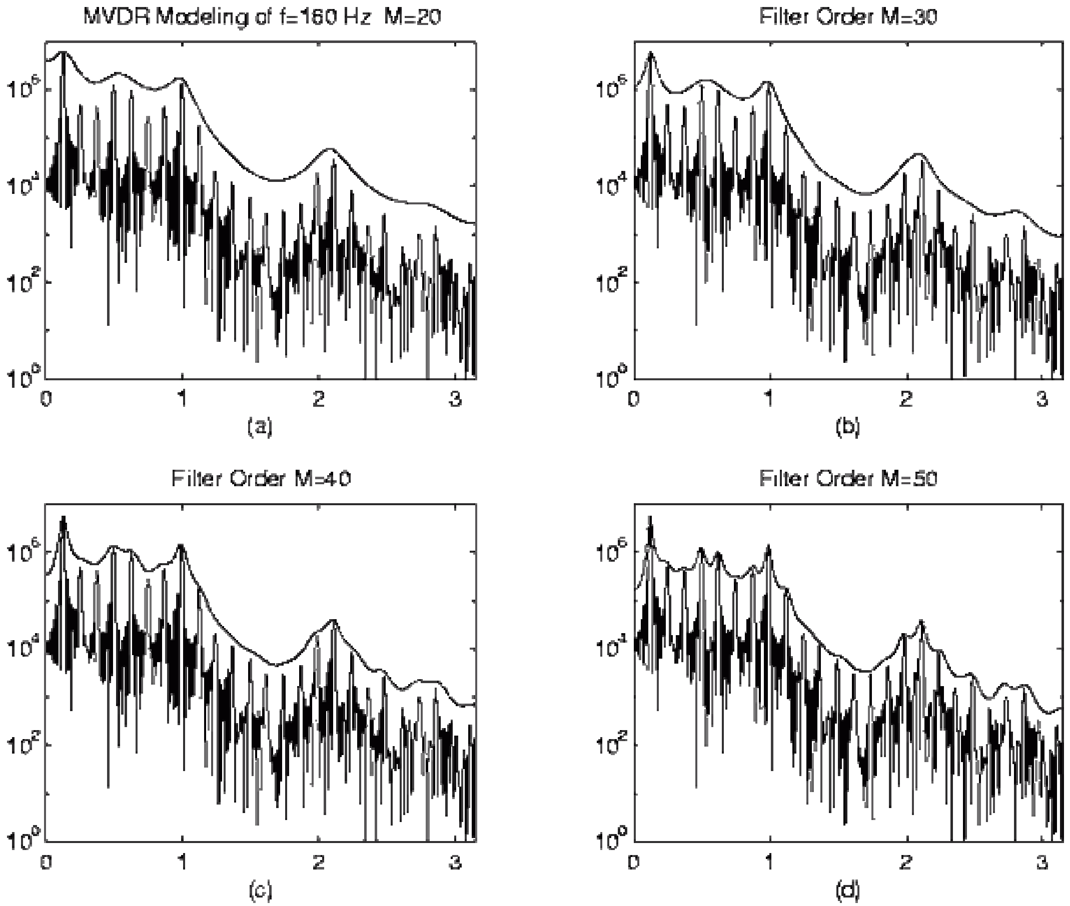
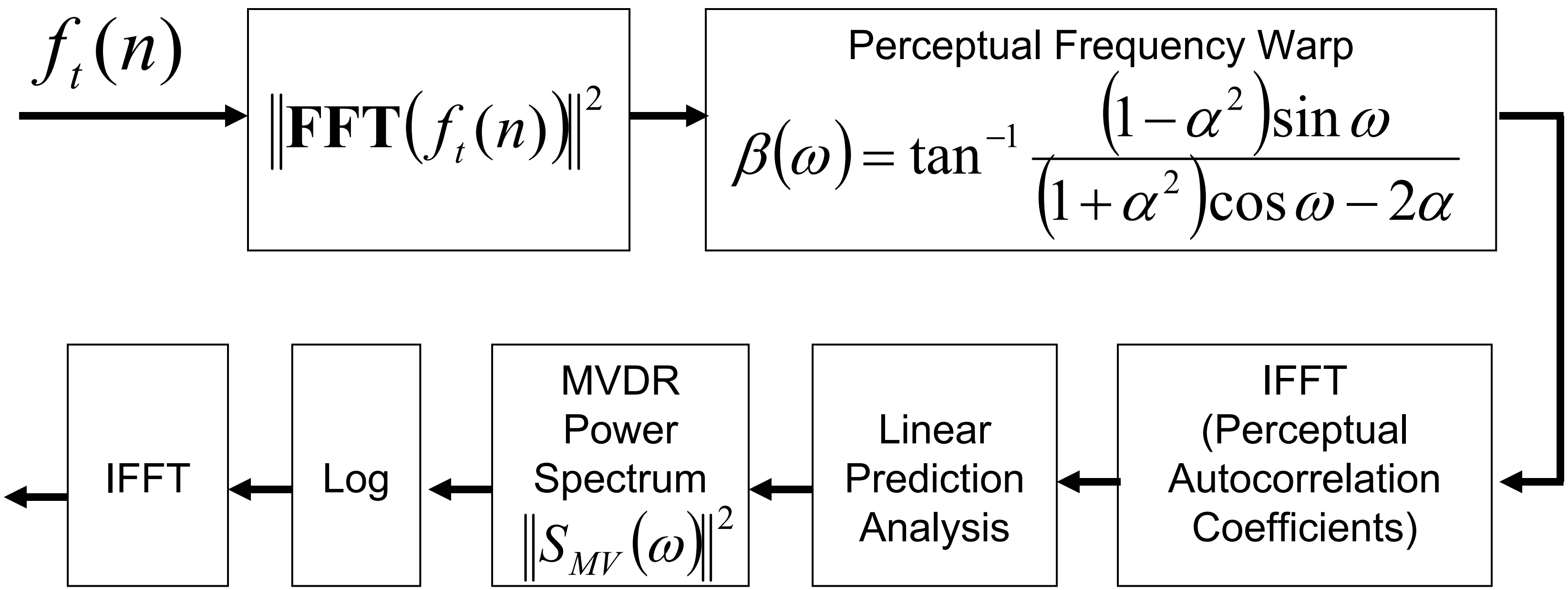


Figure from http://dsp.ucsd.edu/research/speech_coding/

- 160 Hz voiced Speech with model orders M=20,30,40,50

PMVDR Cepstral Coefficients



(Yapanel & Hansen, Eurospeech 2003)

Dynamic Cepstral Coefficients

- Cepstral coefficients do not capture temporal information
- Common to compute velocity and acceleration of cepstral coefficients. For example, for delta (velocity) features,

$$\Delta cep[i][t] = \frac{\sum_{\tau=1}^{D} \tau\left(cep[i][t+\tau] - cep[i][t-\tau]\right)}{2\sum_{\tau=1}^{D} \tau^2}$$

- D typically 2

Dynamic Cepstral Coefficients

- Can also compute the delta-cepstra using the "simple-differences" of the static cepstra,

$$\Delta cep[i][t] = \frac{\left(cep[i][t+D] - cep[i][t-D]\right)}{2D}$$

- D again is typically set to 2.

Frame Energy

- **Frame energy is a typical feature used in speech recognition. Frame energy is computed from the windowed frame,**

$$e[t] = \sum_{m} s^2(n)$$

- **Typically a normalized log energy is used. E.g.,**

$$e_{\max} = \arg\max_{t} \{0.1 \cdot \log(e[t])\}$$

$$E[t] = \arg\max\{-5.0\,,\, 0.1 \cdot \log(e[t]) - e_{\max} + 1.0\}$$

Final Feature Vector for ASR

- **A single feature vector,**
  - 12 cepstral coefficients (PLP, MFCC, …) + 1 norm energy
  - + 13 delta features
  - + 13 delta-delta
- **100 feature vectors per second**
- **Each vector is 39-dimensional**
- **Characterizes the spectral shape of the signal for each time slice**

A few thoughts

- **Current feature extraction methods model each time-slice of the signal as a single shape**
- **Noise at one frequency (a tone) destroys the shape and significantly degrades performance**
- **Human recognition seems to be resilient to localized distortions in frequency…**
- **Several researchers have proposed independent feature streams computed from localized regions in frequency. “Stream-based recognition”.**

Next Time

- Introduction to Hidden Markov Models for Speech Recognition

- Homework #2 due (October 4). See course webpage for details on the assignment. Does not involve any programming or computers this time.