Professor Brendan Morris, SEB 3216, brendan.morris@unlv.edu

EE482: Digital Signal Processing Applications

Speech Signal Processing

http://www.ee.unlv.edu/~b1morris/ee482/

Outline

- Speech Coding
- Speech Enhancement
- Speech Recognition

Speech Coding

- Digital representation of speech signal
  - Provide efficient transmission and storage

- Techniques to compress speech into digital codes and decompress into reconstructed signals
  - Trade-off between speech quality and low bit rate
  - Coding delay and algorithm complexity

Coding Techniques

- Waveform coding
  - Operate on the amplitude of speech signal on per sample basis
- Analysis-by-synthesis coding
  - Process signals by "frame"
  - Achieve higher compression rate by analyzing and coding spectral parameters that represent speech production model
  - Vocoder algorithms transmit coded parameters that are synthesized at receiver into speech

Waveform Coding

- Pulse code modulation (PCM)
  - Simple encoding method by uniform sampling and quantization of speech waveform
- Linear PCM
  - 12-bits/sample for good speech quality
  - 8 kHz sampling rate → 96 kbps
- Non-linear companding (μ-law, A-law)
  - Quantize logarithm of speech signal for lower bit rate → 64 kbps
- Adaptive differential PCM (ADPCM)
  - Use adaptive predictor on speech and quantize difference between speech sample and prediction
  - Lower bit rates because correlation between samples creates good prediction and error signal is smaller amplitude

Linear Predictive Coding (LPC)

- Speech production model with excitation input, gain, and vocal-tract filter

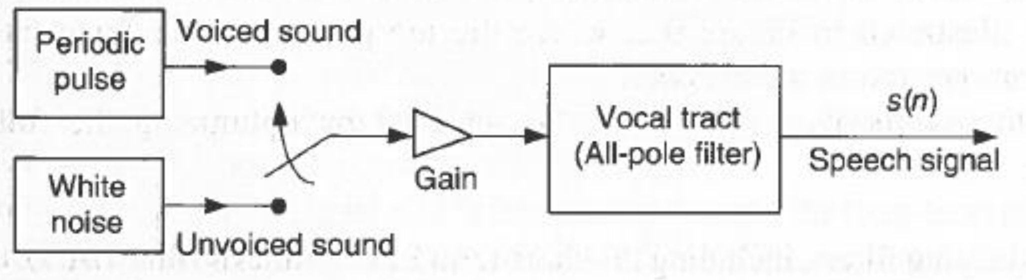


Figure 9.1 Speech production model

  - Vocal tract model is a pipe from vocal cords to oral cavity (with coupled nasal tract)
    - Most important part of model because it changes shape to produce different sounds
    - Based on position of palate, tongue, and lips
- Vocal tract modeled as all pole filter
  - Match a formant (vocal-tract resonance or peaks of spectrum)

(Un)Voiced Sounds

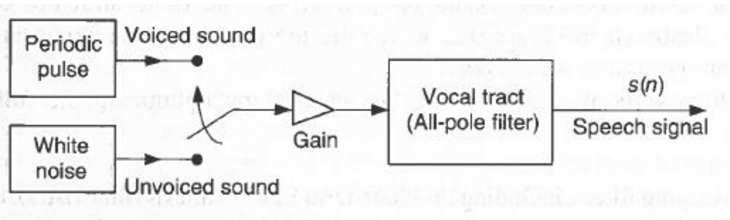


Figure 9.1 Speech production model

- Voiced (e.g. vowels) – caused by vibration of vocal-cords with rate of vibration the pitch
  - Modeled with periodic pulse with fundamental (pitch) frequency
  - Generate periodic pulse train for excitation signal
- Unvoiced (e.g. “s”, “sh”, “f”) – no vibration
  - Use white noise for excitation signal
- Gain represents the amount of air from lungs and the voice loudness

- Speech sounds info [link]

Basic Vocoder Operation

- Process speech in frames

- Usually between 5-30 ms
- Use window function for less ringing
- Windows are overlapped
  - Smaller frame size and higher overlap percentage better captures speech transition → better speech quality

Code-Excited Linear Prediction (CELP)

- Algorithms based on LPC approach using analysis by synthesis scheme
- Coded parameters are analyzed to minimize the perceptually weighted error in synthesized speech
  - Closed-loop optimization with encoder and decoder together
- Optimize three components:
  - Time-varying filters $\{1/A(z), P(z), F(z)\}$
  - Perceptual weighting filter $W(z)$
  - Codebook excitation signal $e_u(n)$

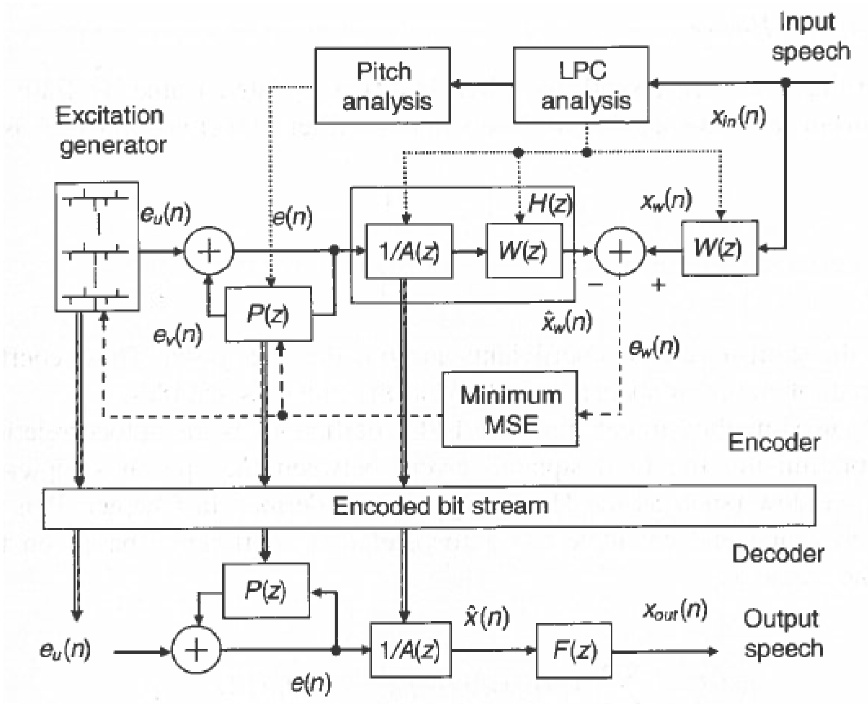


Figure 9.2 Block diagram of typical CELP algorithm

- Notice the excitation, LPC coefficients ($1/A(z)$), and pitch ($P(z)$) coefficients must be encoded and transmitted for decoding and synthesis

Synthesis Filter

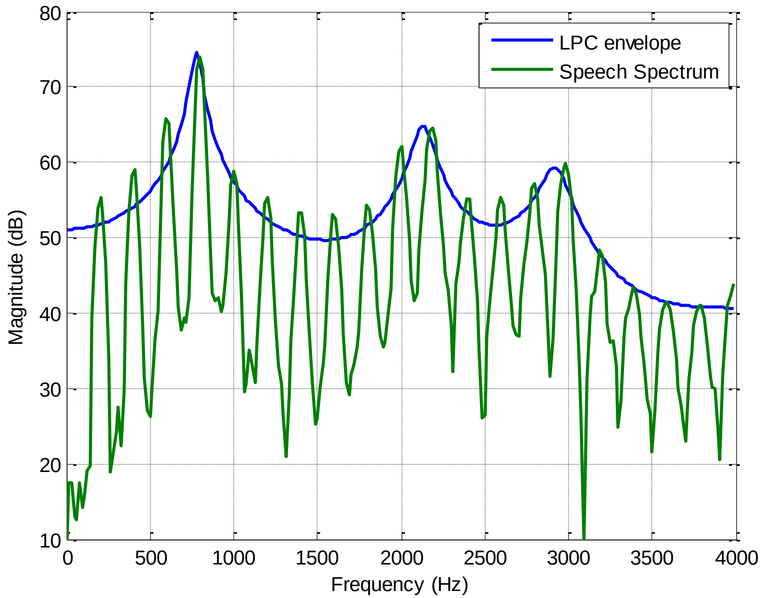
- $1/A(z)$ filter updated each frame with Levinson-Durbin recursive algorithm
  - $\frac{1}{A(z)} = \frac{1}{1-\sum_{i=1}^{p} a_i z^{-i}}$
  - Coefficients used to estimate current speech sample from past samples
- LPC coefficients calculated using autocorrelation method on a frame
  - $r_m(j) = \sum_{n=0}^{N-1-j} x_m(n) x_m(n+j)$
- Solve for LPC coefficients using normal equations

$$\begin{bmatrix} r_m(0) & r_m(1) & \dots & r_m(p-1) \\ r_m(1) & r_m(0) & \dots & r_m(p-2) \\ \vdots & \vdots & \ddots & \vdots \\ r_m(p-1) & r_m(p-2) & \dots & r_m(0) \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_p \end{bmatrix} = \begin{bmatrix} r_m(1) \\ r_m(2) \\ \vdots \\ r_m(p) \end{bmatrix}.$$

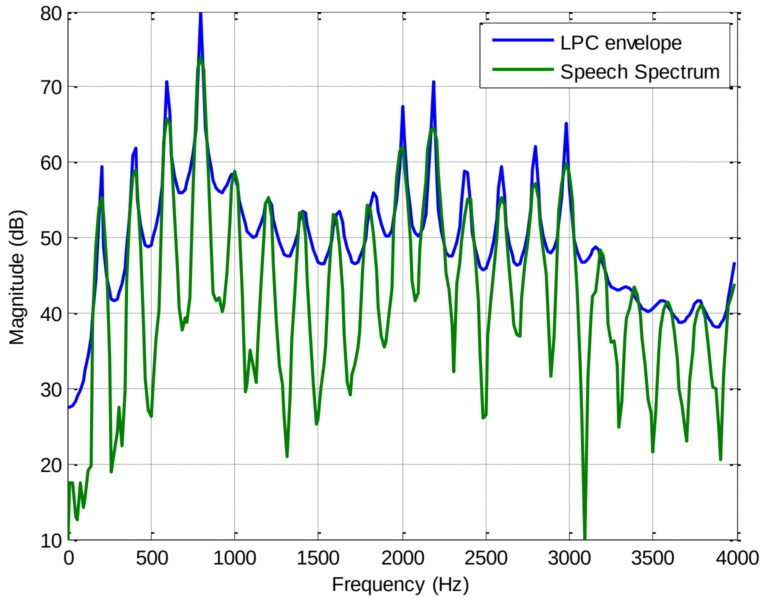
- Can be solved recursively using Levinson-Durbin recursion (pg 334)
  - Matlab `levinson.m` and `lpc.m`

LPC Examples

- Ex 9.2
- Use Levinson-Durbin to estimate LPC coefficients



- Ex 9.3
- Repeat with higher order filter
  - Better match speech spectrum



Excitation Signals

- Short-term – noise signal

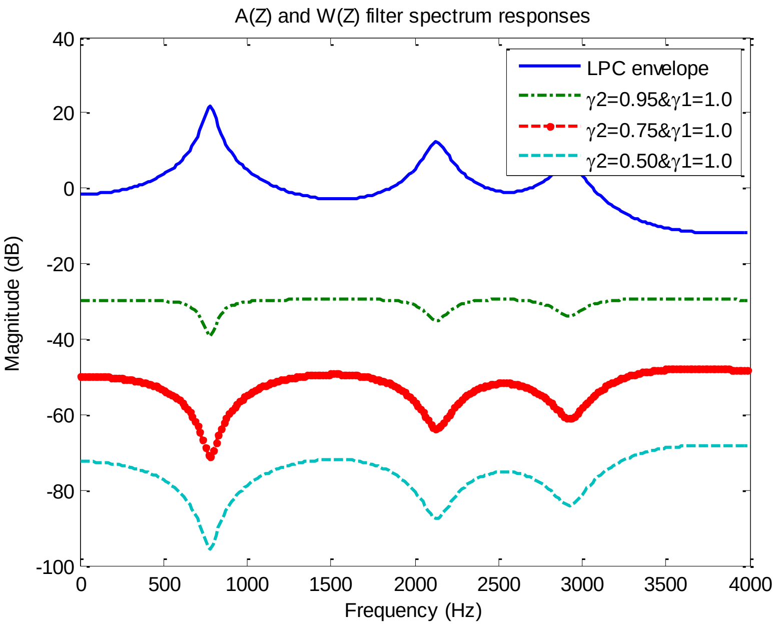
- Long-term – periodic signal
- Pitch synthesis filter models long-term correlation of speech to provide spectral structure
  - $P(z) = \sum_{i=-I}^{I} b_i z^{-(L_{opt}+i)}$
    - L_{opt} - optimum pitch period
- Generally, a frame will be divided into subframes for better temporal analysis
  - Excitation signal is generated per subframe

- An excitation signal is formed as the combination of both short-term and long-term signals
  - $e(n) = e_v(n) + e_u(n)$
    - $e_v(n)$ – voiced long-term prediction excitation
    - $e_u(n)$ – unvoiced noise selected from stochastic codebook (a set of stochastic signals)
- Both excitation signals are passed through $H(z)$ (combined short-term synthesis and perceptual weighting) to find error
  - Will optimize pitch (first) separately from stochastic contribution

Perceptual-Based Minimization

- Perceptual weighting filter $W(z)$ used to control the error calculation
  - Emphasize the weight of errors between format frequencies
    - Shape noise spectrum to place errors in formant regions where humans ears are not sensitive
    - Reduce noise in formant nulls
- $W(z) = \dfrac{A(z/\gamma_1)}{A(z/\gamma_2)}$
  - $\gamma_1 = 0.9, \gamma_2 = 0.5$

- Ex 9.5
- Examine perceptual weighting filter



- Lower γ_2 causes more attenuation at formant frequencies
  - Allows more distortion

Voice Activity Detection (VAD)

- Critical function for speech analysis (for reduction in bandwidth for coding)
- Basic VAD assumptions
  - Spectrum of speech changes in short time but background is relatively stationary
  - Energy level of active speech is higher than background noise

- Practical speech applications highpass filter to remove low-frequency noise
  - Speech is considered in 300 to 1000 Hz range

Simple VAD Algorithm

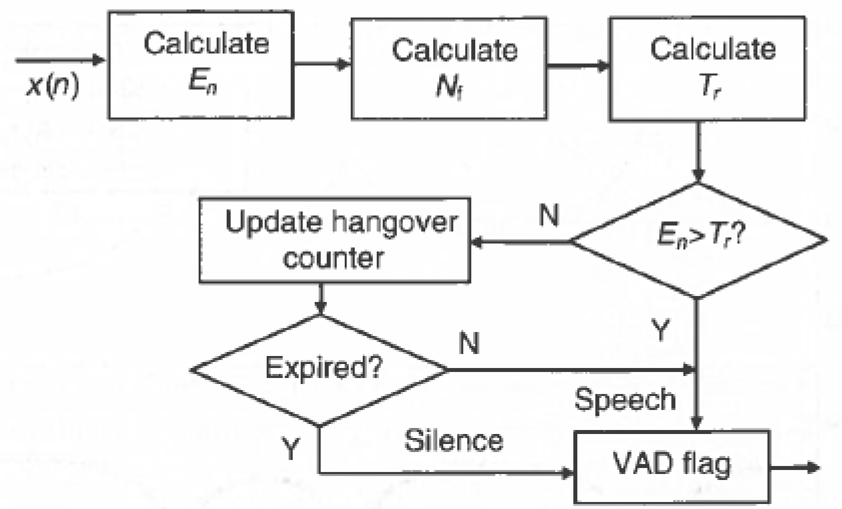


Figure 9.7 Block diagram of simple VAD algorithm

- Calculate frame energy
  - $E_n = \sum_{k=K_1}^{K_2} |X(k)|^2$
    - K_1 bin for 300 Hz
    - K_2 bin for 1000 Hz
  - Recursively compute for short and long windows
- Estimate noise level (floor) N_f
  - Increase noise floor slowly at beginning of speech and quickly at end
- Calculate adaptive threshold
  - $T_r = \frac{N_f}{1-\alpha_l} + \beta$
    - α_l - long window length
    - β – small zero margin
- Threshold signal energy with threshold to determine speech or silence
  - Need a hangover period = 90 ms to handle tail of speech

Speech Enhancement

- Needed because speech may be acquired in a noisy environment
  - Background noise degrades the quality or intelligibility of speech signals
- In addition, signal processing techniques are generally designed under low-noise assumption
  - Degrades performance with noisy environments
- Many speech enhancement algorithms look to reduce noise or suppress specific interference

Noise Reduction

- Will focus on single channel techniques
  - Dual-channel - adaptive noise cancellation from Chapter 6
  - Multi-channel – beamforming and blind source separation
- Three classes:
  - Noise subtraction – subtract estimated amplitude spectrum of noise from noisy signal
  - Harmonic-related suppression – track fundamental frequency with adaptive comb filter to reduce periodic noise
  - Vocoder re-synthesis – estimate speech-model parameters an synthesize noiseless speech

Noise Subtraction

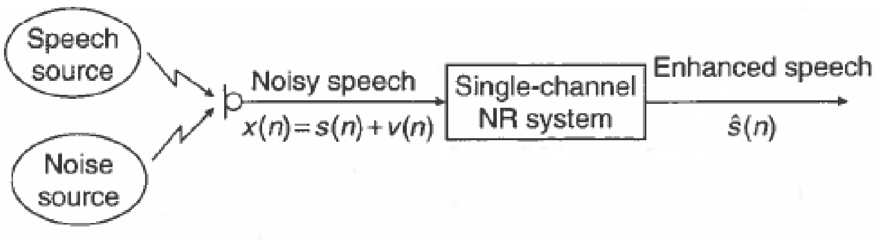


Figure 9.13 A single-channel speech enhancement system

- Input is noisy speech + stationary noise
- Estimate noise characteristics during silent period between utterances
  - Need robust VAD system
- Spectral subtraction – implemented in frequency domain
  - Based on short-time magnitude spectra estimation

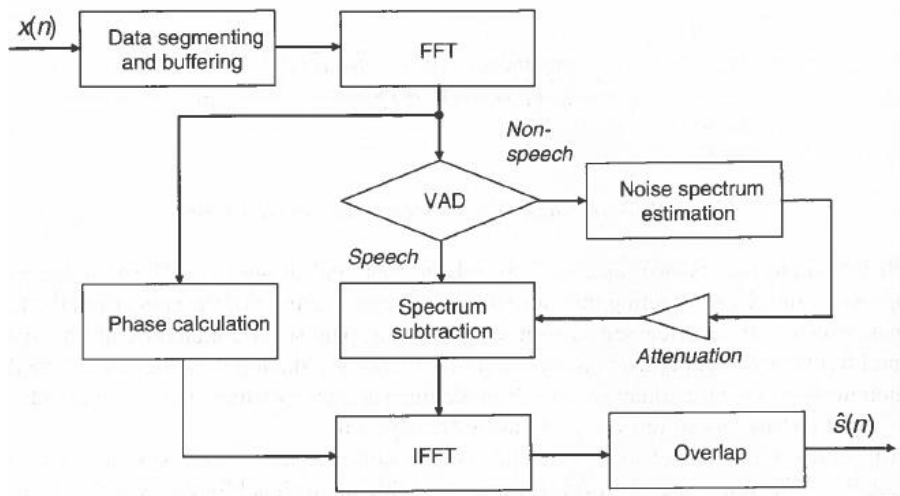


Figure 9.14 Block diagram of the spectral subtraction algorithm

- Subtract estimated noise mag spectrum from input signal
- Reconstruct enhanced speech signal using IFFT
  - Coefficients are difference in mag and original phase

Short-Time Spectrum Estimation

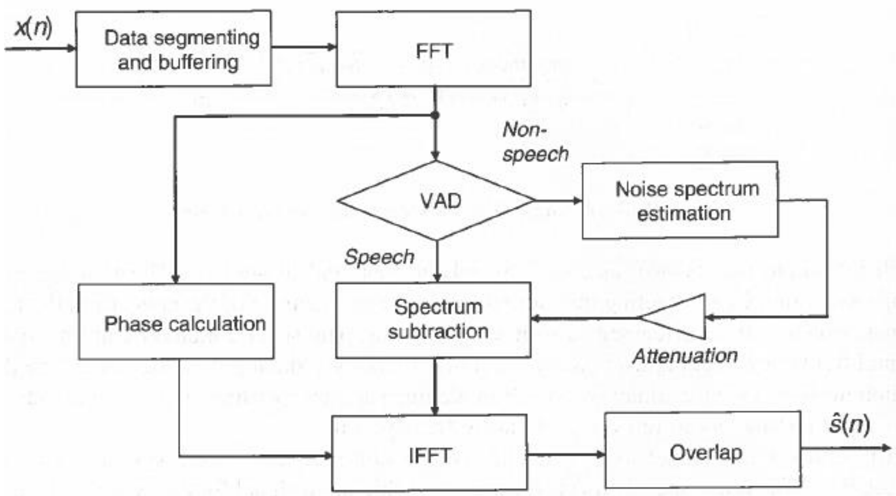


Figure 9.14 Block diagram of the spectral subtraction algorithm

- During non-speech frames, noise spectrum is estimated
- During speech frames, previously estimated noise spectrum is subtracted

- Output for non-speech frames
  - Set frame to zero
  - Attenuate signal by scaling by factor < 1
- Better not to have complete silence in non-speech areas
  - Accentuates noise in speech frames
  - Use 30 dB attenuation

Magnitude Spectrum Subtraction

- Assumes that background noise is stationary an does not change at subsequent frames
- With changing background, algorithm has sufficient time to estimate new noise spectrum

- Modeling noisy speech with noise $v(n)$
  - $x(n) = s(n) + v(n)$
  - $X(k) = S(k) + V(k)$
- Speech estimation
  - $|\hat{S}(k)| = |X(k)| - E|V(k)|$
    - $E|V(k)|$ - estimated noise during non-speech

- Assume human hearing is insensitive to noise in the phase spectrum (only magnitude matters)
- $S(k) = |\hat{S}(k)| \frac{X(k)}{|X(k)|}$
- $S(k) = [|X(k)| - E|V(k)|] \frac{X(k)}{|X(k)|}$
- $S(k) = H(k)X(k)$
  - $H(k) = 1 - \frac{E|V(k)|}{|X(k)|}$

- Notice the phase spectrum never has to be explicitly calculated
  - Avoid computations for arctan

Speech Recognition

- Different than signal processing up to now

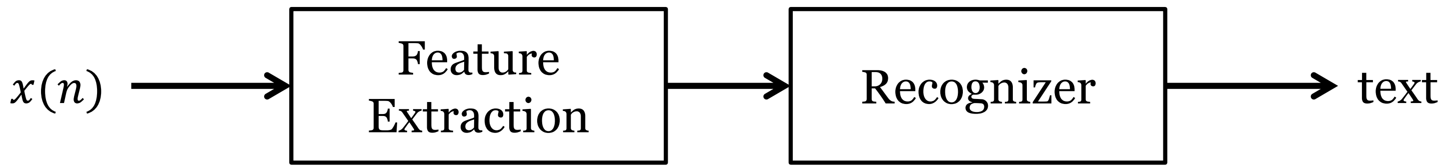
  - Signal input → (enhanced) signal output
- Automatic speech recognition (ASR)

  - Convert speech signal into “text”
    - Label describing speech
- This is a pattern recognition task

ASR Applications and Issues

- Applications
  - Dictation machines
  - Interfaces to devices
  - Reservation systems, phone service, stock quotes, directory assistance
  - Transcribing databases and searching
  - Aids for handicapped
  - Language to language
- Sources of variability in speech
  - Speaker
    - Accent, social context, mood/style, vocal tract size, male/female/child
  - Acoustic environment
    - Background noise reverberation
  - Microphone
    - Non-linear and spectral characteristics
  - Channel
    - Echoes, distortion

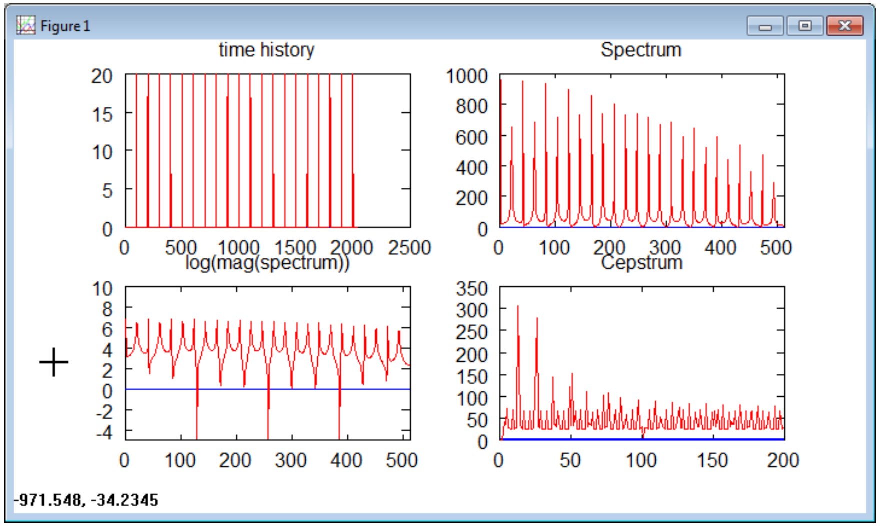
Speech Recognition System



- Feature extraction
  - Represent speech content
  - Typically will use mel-frequency cepstrum (MFCC) coefficients
- Recognizer
  - Pattern recognition system that maps features into text
  - Hidden Markov model (HMM) is popular choice [dynamic time warping (DTW)]
    - See HTK Speech Recognition Toolkit [link]

Cepstrum

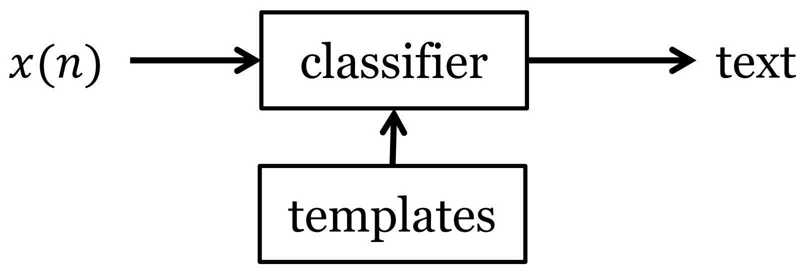
- "Spec"-trum in reverse: "ceps"-strum
- Cepstrum can be seen as information about rate of change in the different spectrum bands
- Calculation:
  - Take FFT: $x(n) \rightarrow X(e^{j\omega})$
  - Take log magnitude: $\log|X(e^{j\omega})|$
  - Take iFFT: $c[n] = \mathcal{F}^{-1}\{\log|X(e^{j\omega})|\}$
- MFCC: Use non-linear frequency bands that mimic human perception
  - Lower frequency have higher resolution



- Using excitation and vocal track model
- $|X(e^{j\omega})| = |H(e^{j\omega})||U(e^{j\omega})|$
- $\log|X(e^{j\omega})| = \log|H(e^{j\omega})| + \log|U(e^{j\omega})|$
- $c_x(n) = c_h(n) + c_u(n)$
  - Can separate excitation from vocal tract with "liftering" (excitation not required for recognition)

Recognition System

- The recognition system is a classifier
  - Compares input speech with a template of known speech to generate output text label



- Templates (reference) patterns
  - $\{R^1, R^2, \ldots, R^V\}$
    - V – size of vocabulary
    - $R^j = \{r_1^j, r_2^j, \ldots, r_{n_j}^j\}$
      - n_j depends on particular template

- Two main tasks:
  - Template design
  - Comparing template with a given observation

- Issues
  - Unequal length data
  - Alignment of speech
  - Distortion (distance) measure for comparison

Log Spectral Distortion

- Given two speech signals $s[n]$ and $s'[n]$
- Log spectral distortion
  - $V(\omega) = \log S(\omega) - \log S'(\omega)$
  - $V(\omega) = \sum (c[n] - c'[n]) e^{-j\omega n}$
  - $d^2(S, S') = \frac{1}{2\pi} \int_{-\pi}^{\pi} |V(\omega)|^2 d\omega$
  - $d^2 = \sum |c[n] - c'[n]|^2$
- Cepstral coefficients as features lead to simple computational procedure
  - $c[0]$ usually not considered in comparison (measure of intensity)
  - Often cepstra derivatives used in representation

Dynamic Time Warping

- Generic method to compare sequences of unequal length
  - Align sequences so that distance is minimized
- Misaligned sequences may be very similar but have large distortion
  - Need alignment to handle different speeds of utterance

- Warping function to align two sequences can be solved efficiently with dynamic program
  - Search for a minimum cost path matching elements of sequences

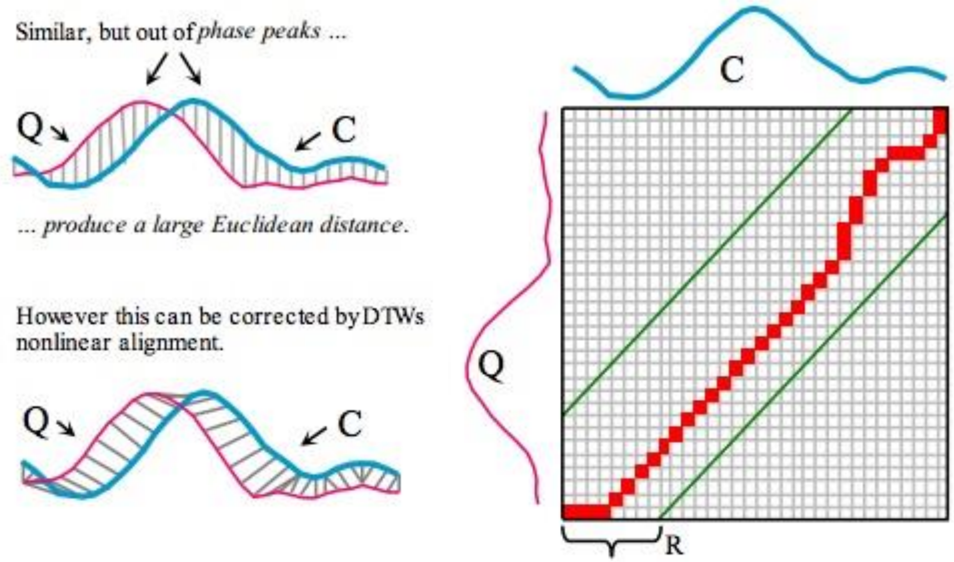


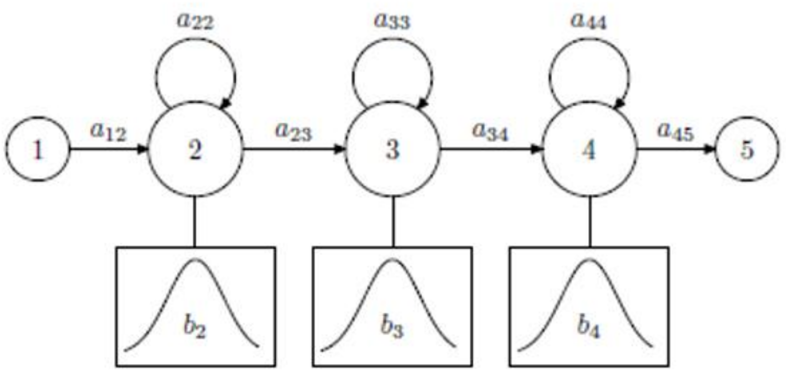
Figure 3: left) Two time series which are similar but out of phase. right) To align the sequences we construct a warping matrix, and search for the optimal warping path (red/solid squares). Note that Sakoe-Chiba Band with width R is used to constrain the warping path

- Each element (cepstrum for a frame) is compared between two sequences to build cost matrix
  - Cost it the distortion between sequence elements

Hidden Markov Models (HMM)

- DTW is restricted to small tasks
  - Cannot include statistical information and to design templates
- HMM is used for statistical model of speech
  - States of HMM correspond to phonemes
  - Don't know state, but observe measurement of state (sound) probabilistically related to state
- Use HMM package

- Use left-to-right HMM



http://www.jmblancocalvo.com/2007/07/speech-recognizer/

- Must learn for each "word":
  - Observation distributions b_i
  - State transitions a_{ij}
- Recognition by evaluating likelihood that a HMM word generated observation $x(n)$