AUTOMATIC SPEECH RECOGNITION

Lecture 7

Word-based Template Matching

This Lecture

- Classification of Speech Recognizer
- Word-based Template Matching
  - Signal Representation
  - Spectral Distance
  - Dynamic Time Warping

Classification of Speech Recognition

- Speaking Style
  - Isolated Word Recognition
  - Continuous Speech Recognition
- Speaker Generalization
  - Speaker-dependent
  - Speaker-independent
- Sound Unit
  - Word-based
  - Sub-word based

Isolated word vs. Continuous speech

- Isolated word recognition
  - one word per utterance
  - system tries to decide the most likely word
- Continuous speech recognition
  - phase or sentence
  - system tries to do:
    - transcription → recognize each word correctly
    - understanding → understand the meaning of the sentence
  - co-articulation among words in the sentence
  - use sophisticated linguistic information
  - apply grammar rules

Speaker-dependent vs. –independent

- Speaker-dependent
  - reference templates/models must be modified every time the speaker changes.
- Speaker-independent
  - system can recognize speech uttered by any speakers
  - much more difficult but necessary in order to broaden the range of possible applications

Sound Units

- The input speech signal is compared with the system's sound units.
- Word-based speech recognizer
  - use 'word' as the smallest unit
  - each word corresponds to one reference template or model
  - no across-word sharing
  - increasing memory size and computation as the number of vocabulary increases
  - cannot handle co-articulation between words in a sentence

Sound Units

- Subword-based speech recognizer
  - The sound units are smaller than 'word'. e.g. phonemes (/a/,/i/,/b/, etc.) or phones ([$b_{closure}$], [$b_{release}$], etc.)
  - Each sub-word unit corresponds to one reference template or model
  - sharing across words
  - The amount of memory size and computation does not greatly increase as the number of vocabulary increases.

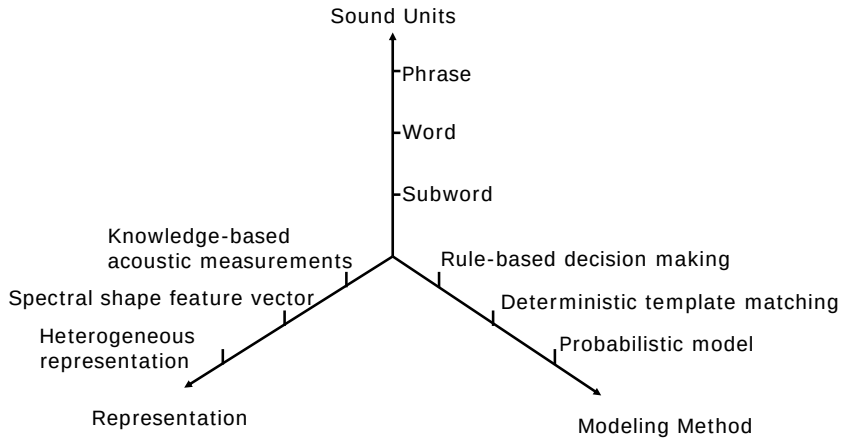
Sound Units

- Context-dependent subword units
  - context-dependent subword units are necessary for handling of co-articulation
  - Two sounds with the same phoneme but different surrounded phonemes are considered different sounds. (Therefore, different models have to be constructed.)
  - "***Triphone***" → a specific phoneme with specific left- and right- contexts.
    - สาม → */sil+s-a4/ /s+a4-m^/ /a4+m^-sil/*
    - ขาม → */sil+kh-a4/ /kh+a4-m^/ /a4+m^-sil/*
    - ขาน → */sil+kh-a4/ /kh+a4-n^/ /a4+n^-sil/*

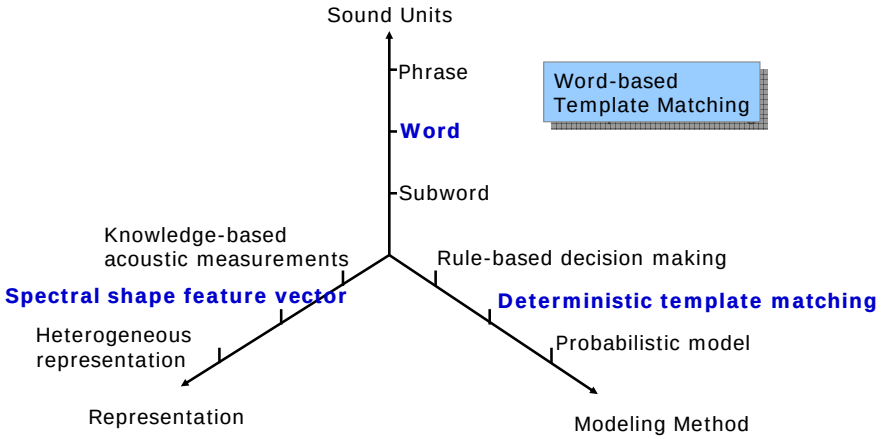
Sound Units

- The most appropriate units for enabling speech recognition success depends on:
  - type of recognition task
  - vocabulary size
- Some medium-sized subword units between words and phonemes have also been explored in order to overcome the disadvantages of using either words or phonemes.
  - e.g. demisyllables, onsets/rhymes

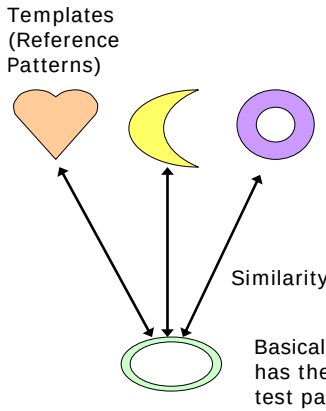
Acoustic Modeling Approaches



Acoustic Modeling Approaches



Template Matching



Issues:

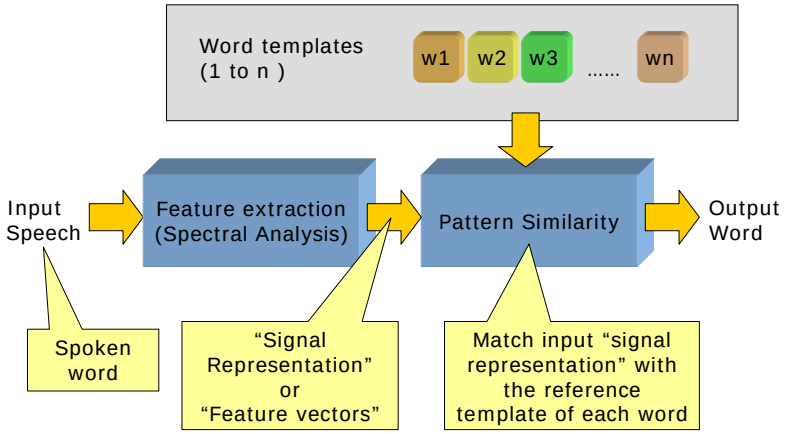
What should be the representation used in the patterns?

What should be the measure of similarity?

How can we match patterns?

Basically, pick the template that has the best similarity to the test pattern.

Isolated-word recognition system



Issues:

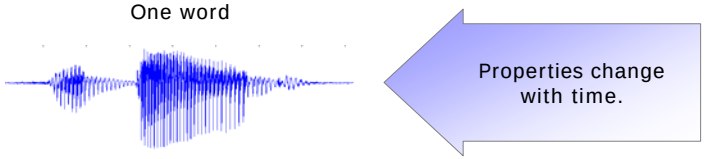
What should be the representation used in the patterns?

What should be the measure of similarity?

How can we match patterns?

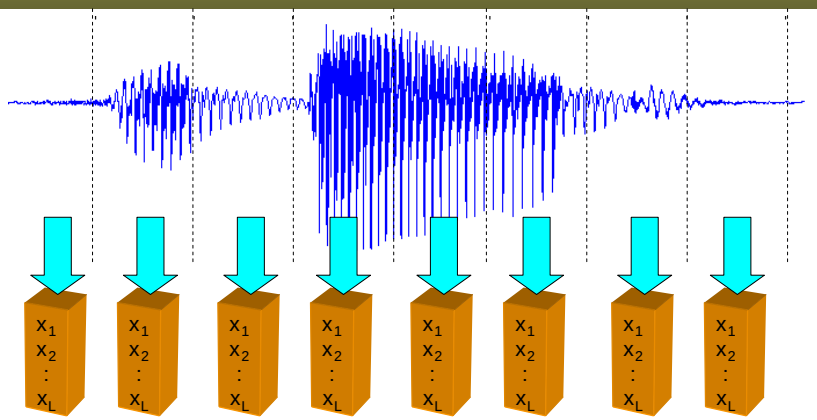
Word-based Template Matching

- What should be the representation used in the patterns?



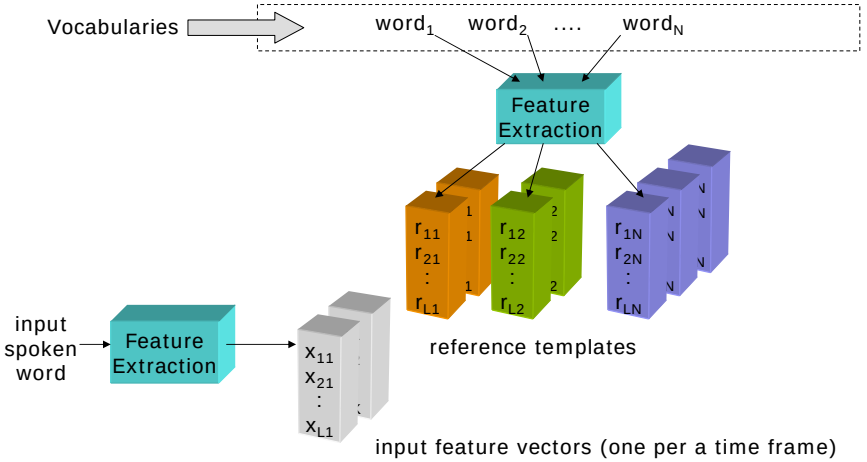
We must extract series of “features”. Each for a time frame. So the features in each time frame captures rather stable properties.

Frame-based Feature Extraction



feature extraction is performed on a frame-based basis → one vector/time frame

Feature Extraction



Feature Extraction

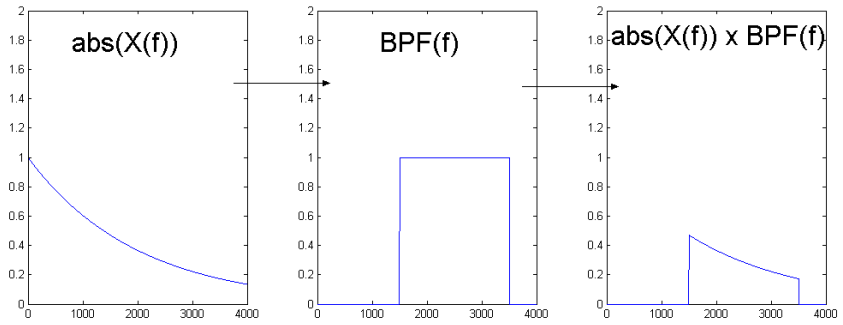
- In most speech recognition system, short-time spectral similarities between input speech and reference units are calculated as the basis for the recognition decision.
- The process of extracting these spectral characteristics of the input speech is referred to as "Feature extraction".
- These spectral features (or signal representations) come in many forms, e.g.: FFT coef., Cepstral coef., LPC etc.

Signal Representation

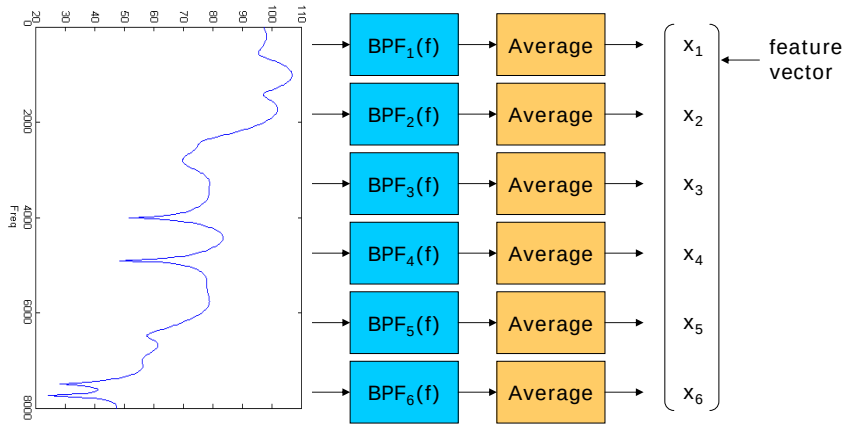
- Each dimension of a 'feature vector' might be from the following spectral analyses:
  - Output from band-pass filter bank
  - Discrete Fourier Transform (implement using FFT)
  - Linear Predictive Coding (LPC)
  - Mel Frequency Cepstral Coefficients (MFCC)
  - Auditory Features

Band-pass Filter Bank

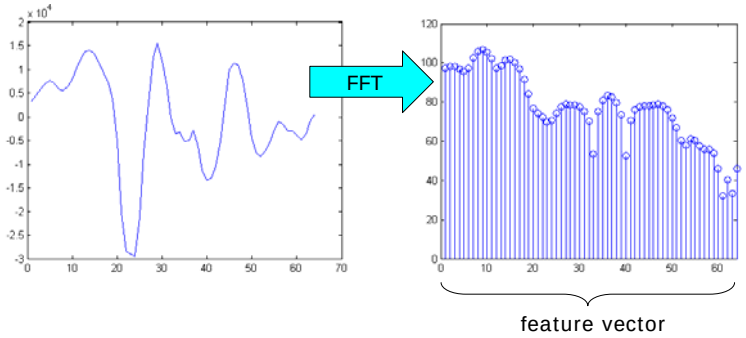
- Band-pass filter
  - Let certain frequency range through



Band-pass Filter Bank



Discrete Fourier Transform (using FFT)



Spectral fine structure varies according to pitch and many factors

Linear Predictive Coding (LPC)

- In this method, speech spectrum is modeled by the position and amplitude of spectral peaks.
- A small number of parameters (LPC coefficients) are used for reconstruction of speech signal, usually 2 LPC coeff. per peak.
- These coefficients are used as a type of feature vector in the ASR problem.

Mel Frequency Cepstral Coefficients (MFCC)

- Cepstral analysis is a common procedure used for separating speech signal into source and vocal tract filter.
- Low-order cepstral coefficients capture the characteristics of the vocal tract when the speaker utters that sound.
- 13 cepstral coeff + 13 first order derivatives and 13 second order derivative are a popular feature vector for ASR
- Mel-scale is used to mimic auditory processing
- typically out-performs Fourier- and LPC-based.

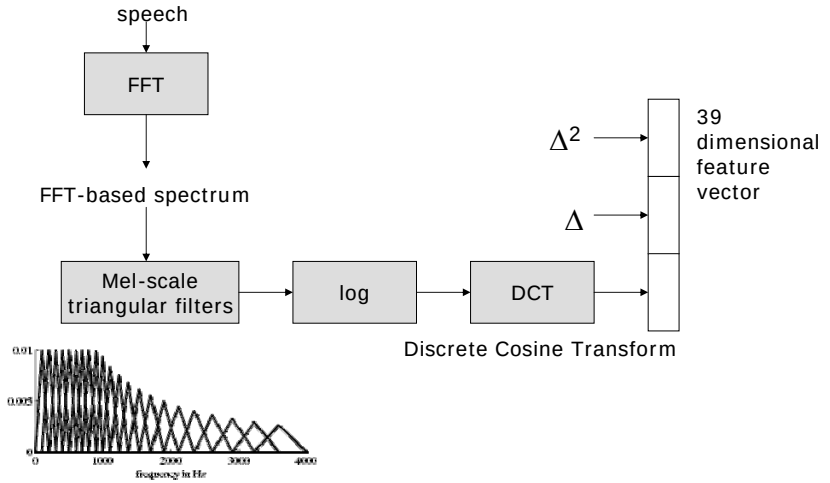
Mel Frequency Cepstral Coefficients (MFCC)

MFCCs are commonly derived from these steps:

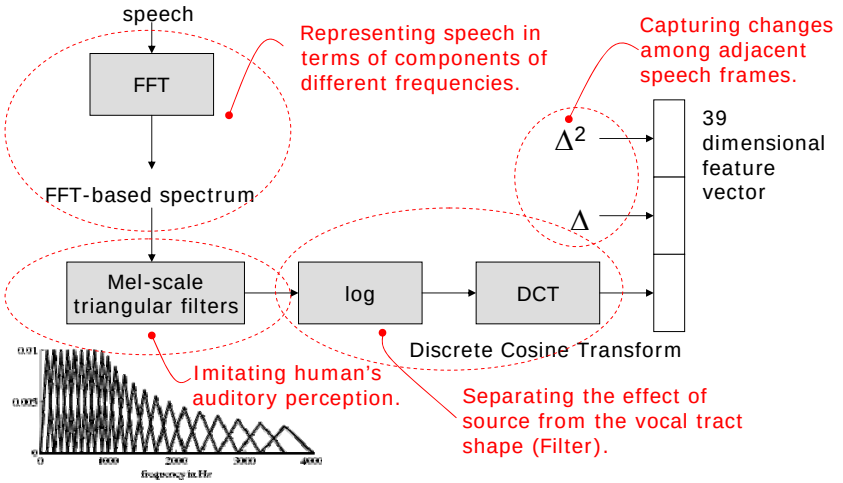
1. Take the Fourier transform of (a windowed excerpt of) a signal.
2. Map the powers of the spectrum obtained above onto the mel scale, using triangular overlapping windows.
3. Take the logs of the powers at each of the mel frequencies.
4. Take the discrete cosine transform of the list of mel log powers, as if it were a signal.

The MFCCs are the amplitudes of the resulting spectrum.

MFCC



MFCC



Articulatory features

- Each dimension in the articulatory feature vector is a measurement that intends to capture the underlying speech production mechanism directly.
- for example:
  - Formant frequencies
  - Pitch (fundamental frequency)
- Use extensive understanding of human speech production
- Usually hard to determine correctly e.g. formant detection

Issues:

What should be the representation used in the patterns?

What should be the measure of similarity?

How can we match patterns?

Spectral Distance

- Comparing similarity among feature vectors

Measures of Spectral Similarity

- The shorter the distances, the more similar the feature vectors are.
- Various distance measures can be defined based on multivariate vectors representing short-time spectra which are obtained through the mentioned techniques.

Spectral Distance

- The distance measure, d(x,y), must satisfy:
  - Symmetry:
    - » d(x,y) = d(y,x)
  - Positive Definiteness
    - » d(x,y) > 0; x ≠ y
    - » d(x,y) = 0; x = y

Euclidean Distance

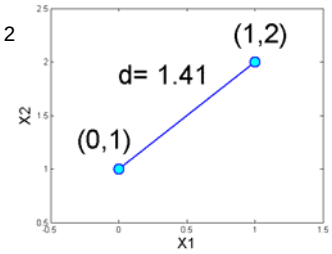
- Euclidean Distance between X and Y

$$X = (x_1, x_2, ..., x_p), Y = (y_1, y_2, ..., y_p)$$

$$d(X,Y) = \sqrt{\sum_{i=1}^{p} (x_i - y_i)^2}$$

Every dimension has similar significance

p=2



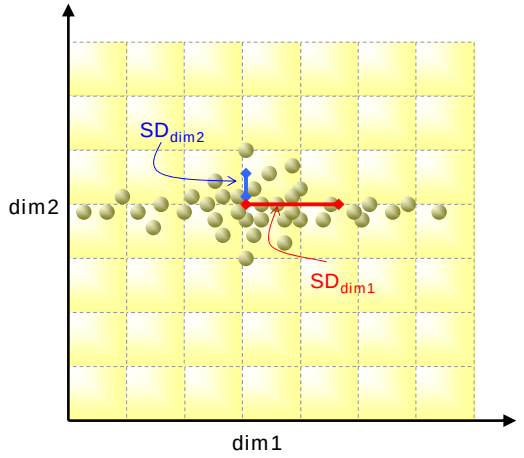
Weighted Distance

- Each dimension has a weight.

$$d(X,Y)=\sqrt{\sum_{i=1}^{p}(w_i(x_i-y_i))^2}$$

- bigger w_i for important significant dimension
- e.g.:
  - Mahalanobis Distance: each dimension is weighted with the weight that is inversely proportional to the variance in that direction.

Weighted Distance: Mahalanobis



1 unit of "dim1" does not seem as significant as 1 unit of "dim2".

Issues:

What should be the representation used in the patterns?

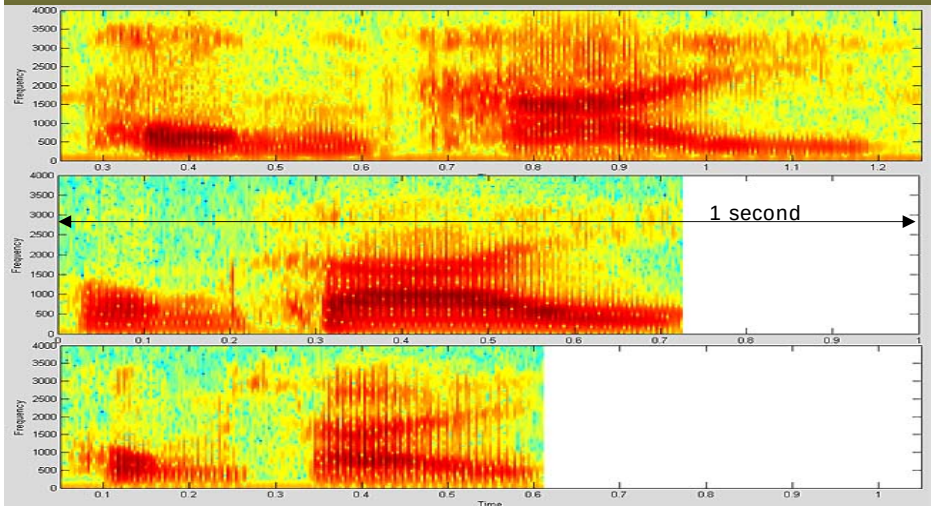
What should be the measure of similarity?

How can we match patterns?

Template Matching Mechanism

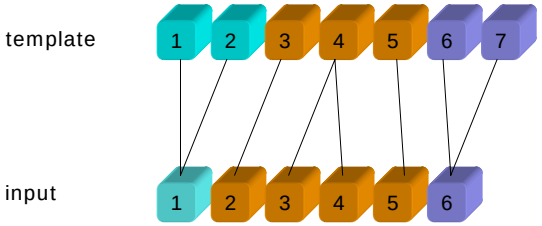
- Test pattern, T ,and reference pattern, R, are represented by sequences of feature measurements.
  - $T=\{t_1,t_2,...,t_{Kt}\}$
  - $R=\{r_1,r_2,...,r_N\}$
- Pattern similarity is determined by aligning test pattern, T, with reference pattern, R_n, with distance $D(T,R_n)$
- Decision rule chooses reference pattern, R^*, with smallest distance $D(T,R^*)$
  - $R^* = arg\ min_n\ D(T, R_n)$

Aligning Problem

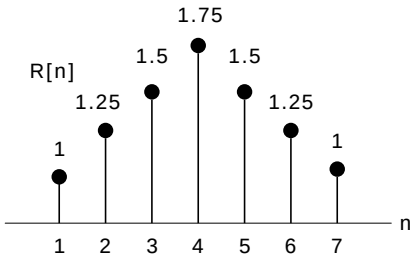


Dynamic Time Warping

- Dynamic Time Warping (DTW) is used to compute the best possible alignment warp between T and R_v, and the associated distortion $D(T, R_v)$
- Non-linearly expands or contracts the time axis to match the input speech with reference templates.



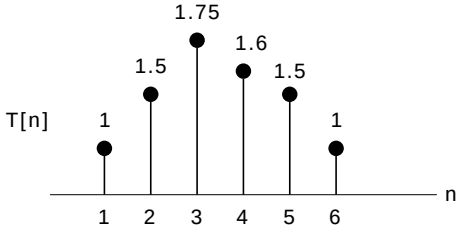
1-D aligning example



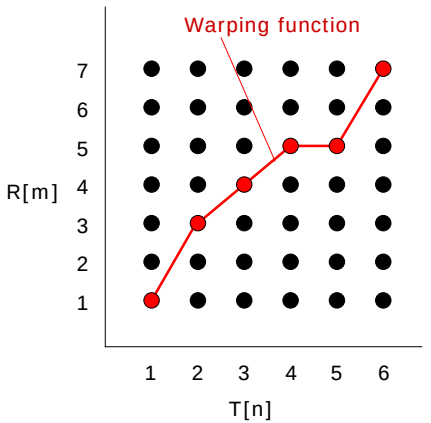
Distance (x,y) = |x-y|

(1-D Euclidean Distance)

Find alignment that minimize the total distance



1-D aligning example



$$R = \begin{pmatrix} 1 \\ 1.25 \\ 1.5 \\ 1.75 \\ 1.5 \\ 1.25 \\ 1 \end{pmatrix} \quad T = \begin{pmatrix} 1 \\ 1.5 \\ 1.75 \\ 1.6 \\ 1.5 \\ 1 \end{pmatrix}$$

Total Distance = |1-1|
+|1.5-1.5|
+|1.75-1.75|
+|1.5-1.6|
+|1-1|

= 0.1

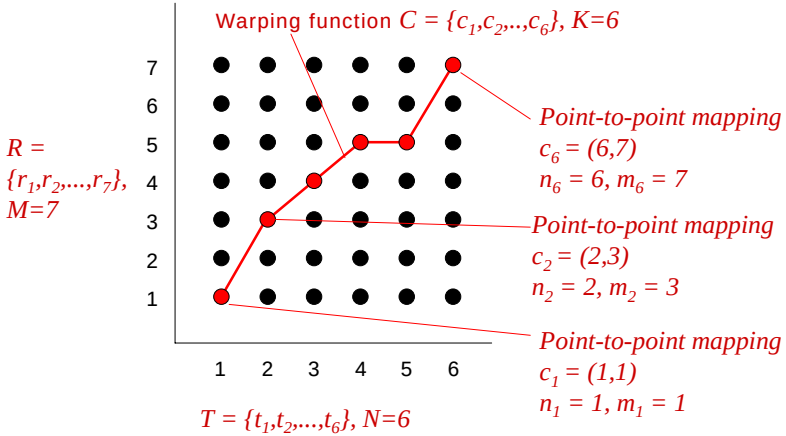
Dynamic Programming for DTW

- A popular algorithm to find the best path (mapping) through the plane spanned by T and R is to use Dynamic Programming (DP Matching)

DP Matching: Objective and Notations

- Objective: find an optimal alignment between variable length feature vectors $T=\{t_1,...,t_N\}$ and $R=\{r_1,...,r_M\}$
- The overall distance $D(T,R)$ is based on sum of local distances between element $d(t_i,r_j)$
- A warping function, C, aligns T and R via a point-to-point mapping.
  - $C = \{c_1,..,c_K\}$ when $c_k = (n_k,m_k)$
  - $c = (n,m)$ is a point on the plane spanned by T and R

DP Matching: Notations used



DP Matching: Optimization Criteria

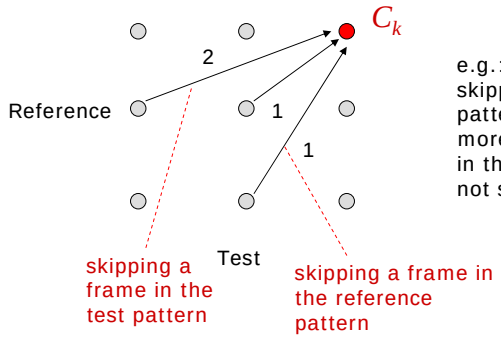
- minimize the sum of the distance between two feature vectors along C, which is represented by:

$$D(C) = \frac{\sum_{k=1}^{K} d(c_k) w_k}{\sum_{k=1}^{K} w_k}$$

- w_k is a positive weighting function related to C

DP Matching: Weighting function

- w_k is the weight for the transition from any C_{k-1} to C_k



e.g.:
skipping frame in the test pattern is weighted twice more than skipping a frame in the reference pattern or not skipping a frame

DP Matching: Conditions

- Minimize D(C) under the following conditions:

Monotony and continuity condition

$0 \le n_k - n_{k-1},\ 0 \le m_k - m_{k-1}$
(moving forward in time for both vectors)

Boundary condition

$n_1 = m_1 = 1,\ n_K = N,\ m_K = M$
or
$c_1 = (1,1),\ c_K = (N,M)$

DP Matching: weight selection condition

- If $\sum_{k=1}^{K} w_k = W_c$ is constant,

$$D(C) = \frac{\sum_{k=1}^{K} d(c_k) w_k}{\sum_{k=1}^{K} w_k} = \frac{1}{W_c} \sum_{k=1}^{K} d(c_k) w_k$$

- then

$$\arg\min_{c1,c2,\ldots,cK} \frac{1}{W_c} \sum_{k=1}^{K} d(c_k) w_k = \arg\min_{c1,c2,\ldots,cK} \sum_{k=1}^{K} d(c_k) w_k$$

DP Matching: Algorithm

$$g(c_k) = \min_{c1,c2,\ldots,ck-1} [\sum_{i=1}^{k} d(c_i) w_i]$$

minimum total distance of sub-path {c1, c2, ..., ck}

$$= \min_{c1,c2,\ldots,ck-1} [\sum_{i=1}^{k-1} d(c_i) w_i + d(c_k) w_k]$$

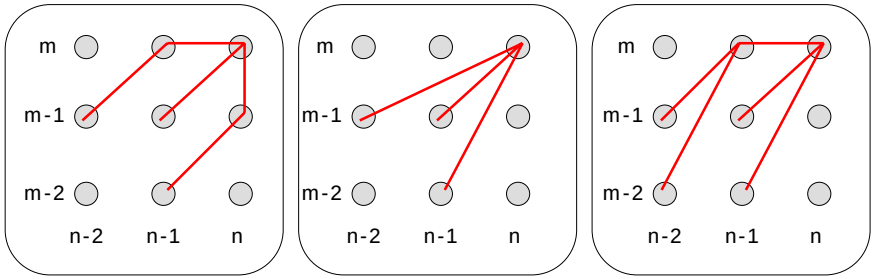
$$= \min_{ck-1} [\min_{c1,c2,\ldots,ck-2} \{\sum_{i=1}^{k-1} d(c_i) w_i\} + d(c_k) w_k]$$

$$= \min_{ck-1} [g(c_{k-1}) + d(c_k) w_k]$$

avoid exhaustively searching through all possibilities for C

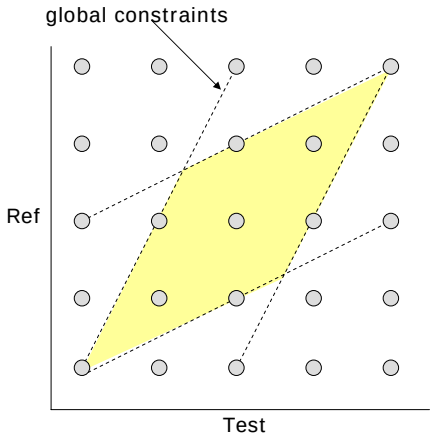
$$g(c_k) = \min_{ck-1} [g(c_{k-1}) + d(c_k) w_k]$$

DP Matching: Local constraints



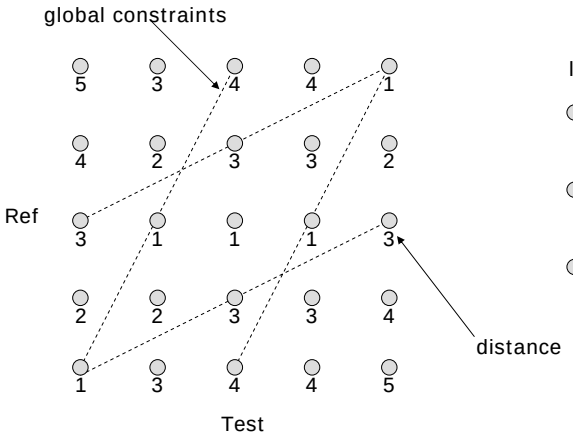
Local constraints determine alignment flexibility

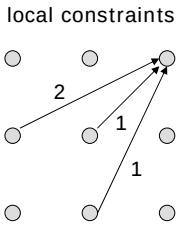
DP Matching: Global constraints



Global constraints exclude portion of search space

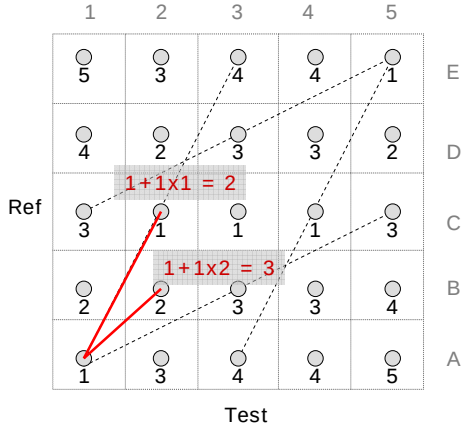
DP Matching: Example

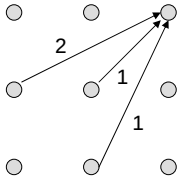




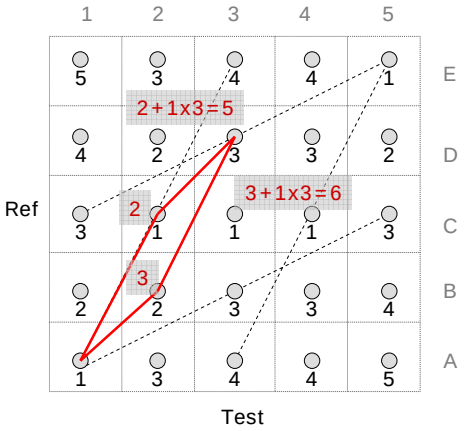
Find the best warping

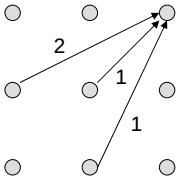
DP Matching: Example





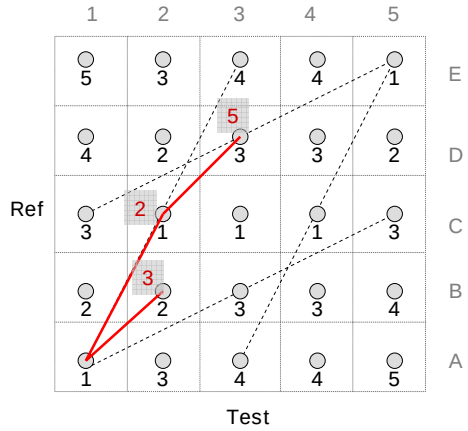
DP Matching: Example

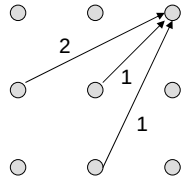




Department of Computer Engineering, Chulalongkorn University
Spoken Language Systems Research Group

DP Matching: Example

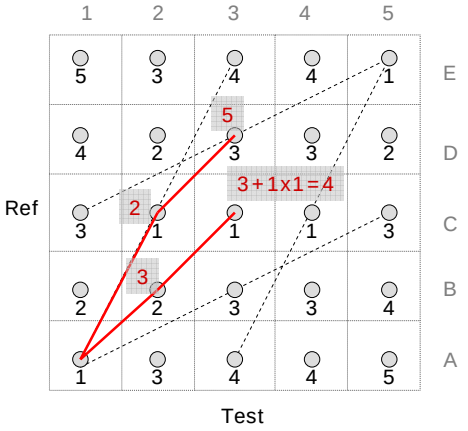


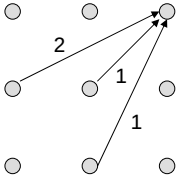


Special Topics in Computer Science | First Semester 2008 | Lecture 7
ATIWONG SUCHATO

Department of Computer Engineering, Chulalongkorn University
Spoken Language Systems Research Group

DP Matching: Example

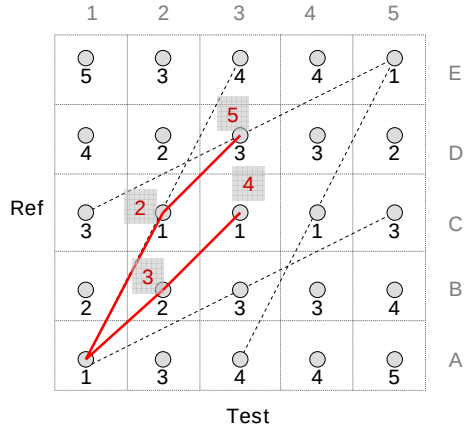


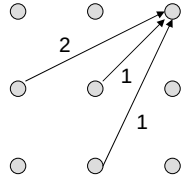


Special Topics in Computer Science | First Semester 2008 | Lecture 7
ATIWONG SUCHATO

Department of Computer Engineering, Chulalongkorn University
Spoken Language Systems Research Group

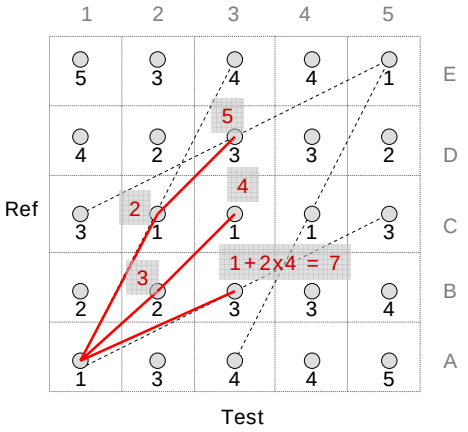
DP Matching: Example

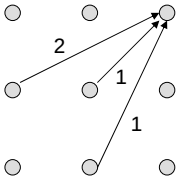




Special Topics in Computer Science | First Semester 2008 | Lecture 7
ATIWONG SUCHATO

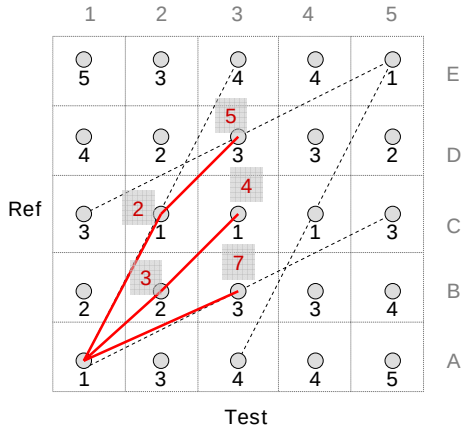
DP Matching: Example

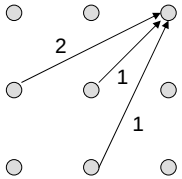




Department of Computer Engineering, Chulalongkorn University
Spoken Language Systems Research Group

DP Matching: Example

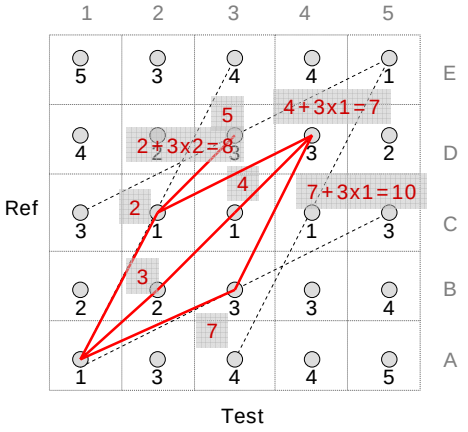


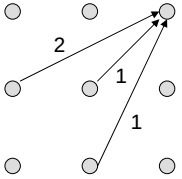


Special Topics in Computer Science | First Semester 2008 | Lecture 7
ATIWONG SUCHATO

Department of Computer Engineering, Chulalongkorn University
Spoken Language Systems Research Group

DP Matching: Example

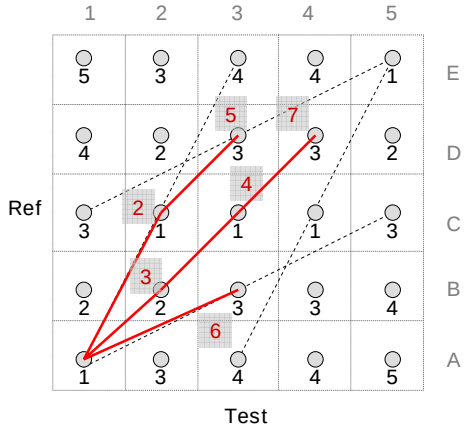


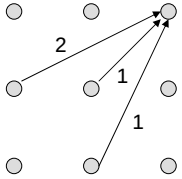


Special Topics in Computer Science | First Semester 2008 | Lecture 7
ATIWONG SUCHATO

Department of Computer Engineering, Chulalongkorn University
Spoken Language Systems Research Group

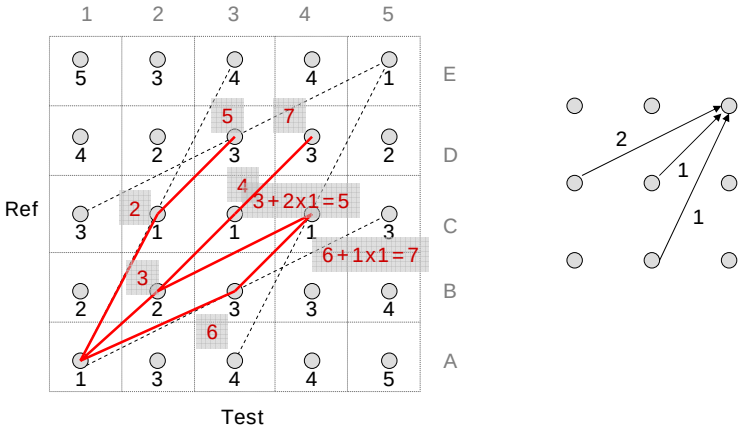
DP Matching: Example





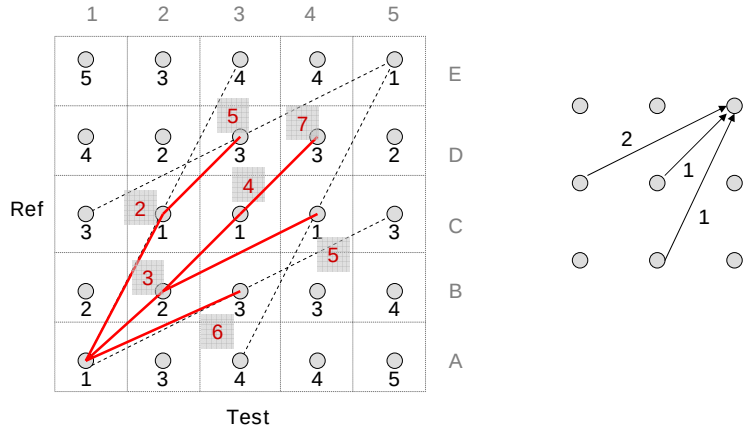
Special Topics in Computer Science | First Semester 2008 | Lecture 7
ATIWONG SUCHATO

DP Matching: Example



Department of Computer Engineering, Chulalongkorn University
Spoken Language Systems Research Group

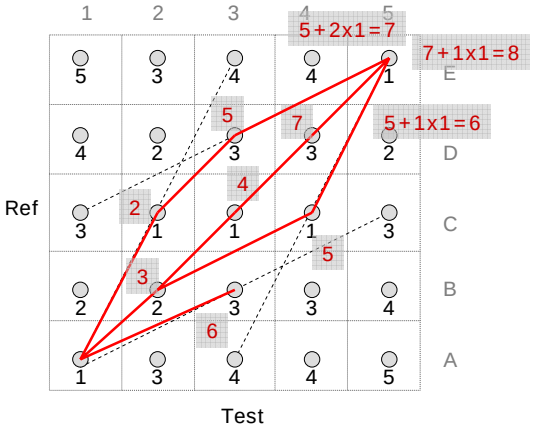
DP Matching: Example

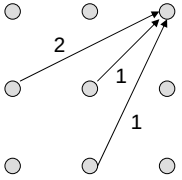


Special Topics in Computer Science | First Semester 2008 | Lecture 7
ATIWONG SUCHATO

Department of Computer Engineering, Chulalongkorn University
Spoken Language Systems Research Group

DP Matching: Example

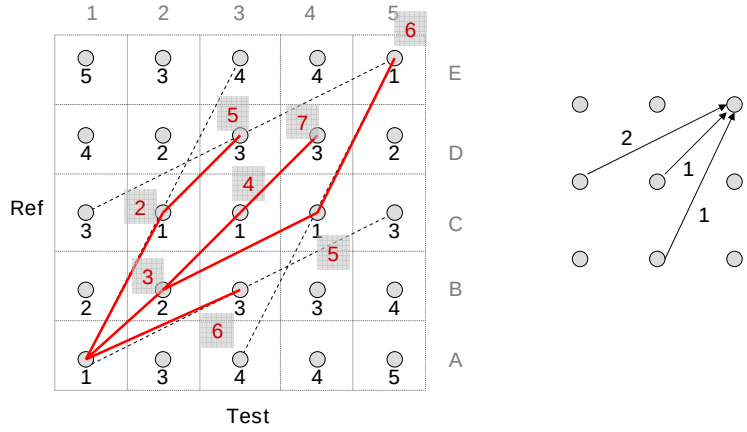




Special Topics in Computer Science | First Semester 2008 | Lecture 7
ATIWONG SUCHATO

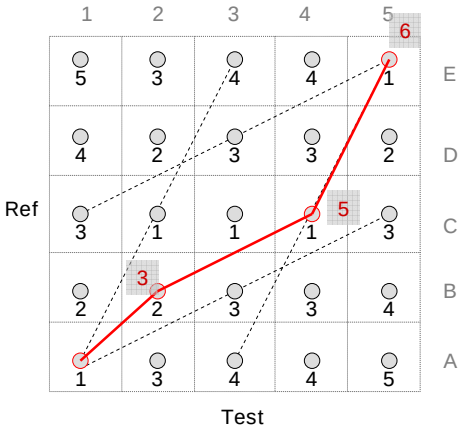
Department of Computer Engineering, Chulalongkorn University
Spoken Language Systems Research Group

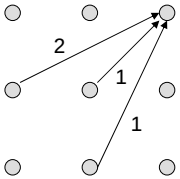
DP Matching: Example



Special Topics in Computer Science | First Semester 2008 | Lecture 7
ATIWONG SUCHATO

DP Matching: Example





Word-based Template Matching Summary