CH8: TEMPLATE MATCHING

- The Goal: Given a set of reference patterns known as TEMPLATES, find to which one an unknown pattern matches best. That is, each class is represented by a single typical pattern.

- The crucial point is to adopt an appropriate "measure" to quantify similarity or matching.

- These measures must accommodate, in an efficient way, deviations between the template and the test pattern. For example the word beauty may have been read a beeauty or beuty, etc., due to errors.

- Typical Applications
  - Speech Recognition
  - Motion Estimation in Video Coding
  - Data Base Image Retrieval
  - Written Word Recognition
  - Bioinformatis
  - …
- Measures based on optimal path searching techniques
  - Representation: Represent the template by a sequence of measurement vectors

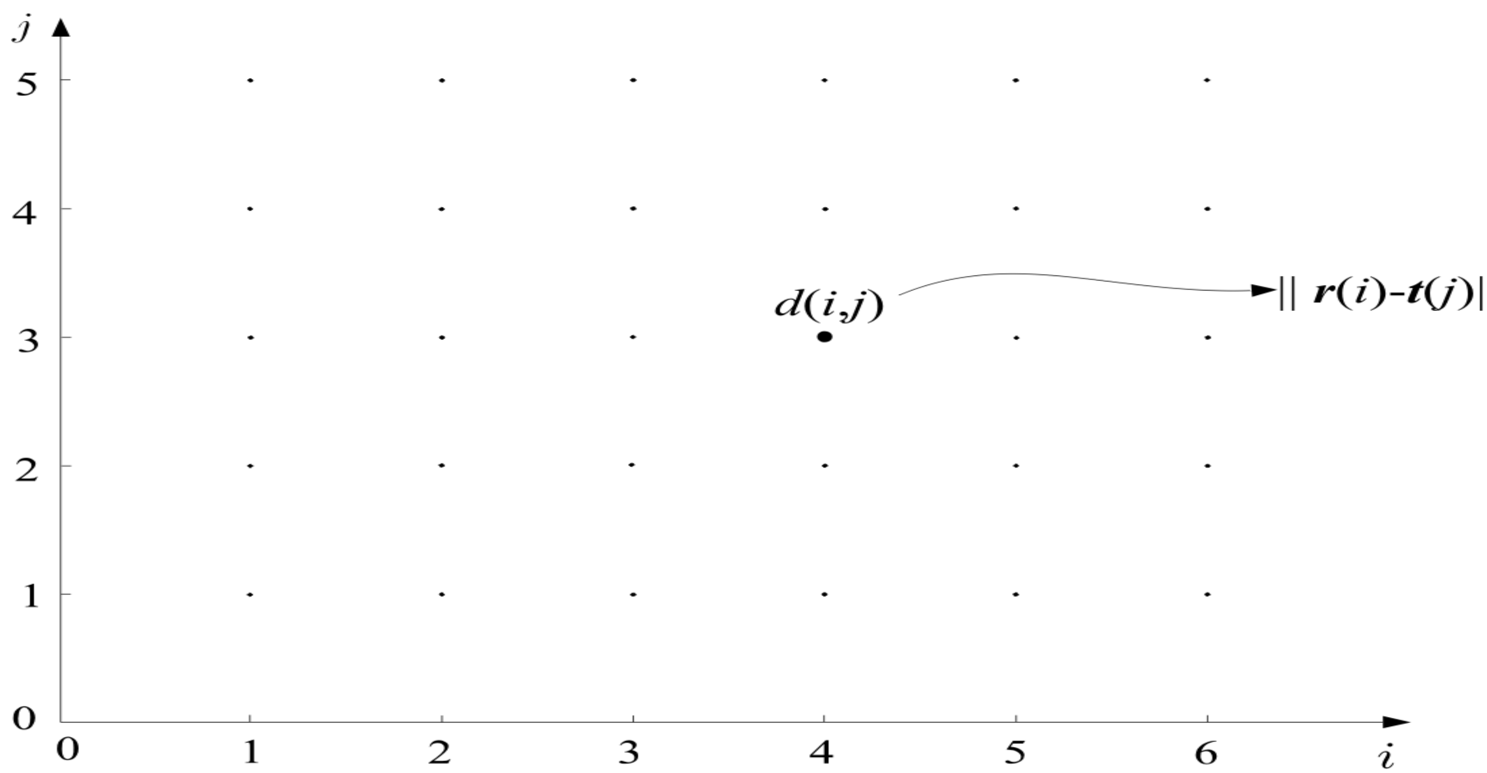
Template: $\underline{r}(1), \underline{r}(2), \ldots, \underline{r}(I)$

Test pattern: $\underline{t}(1), \underline{t}(2), \ldots, \underline{t}(J)$

- In general $I \neq J$

- Form a grid with I points (template) in horizontal (abscissa) and J points (test) in vertical (ordinate)

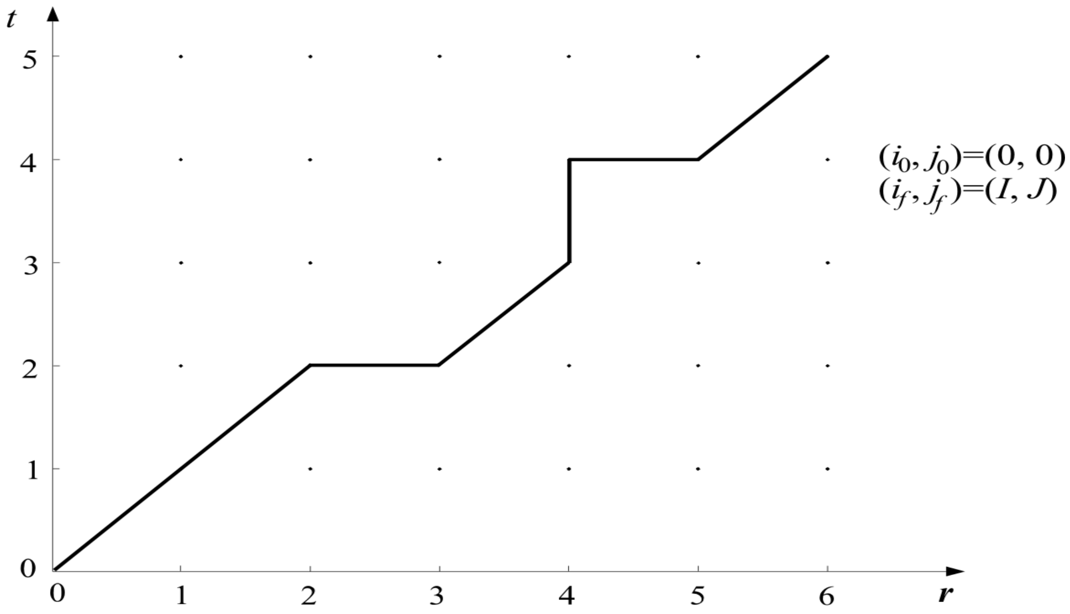
- Each point (i,j) of the grid measures the distance between $\underline{r}(i)$ and $\underline{t}(j)$



- Path: A path through the grid, from an initial node (i_0, j_0) to a final one (i_f, j_f), is an ordered set of nodes $(i_0, j_0), (i_1, j_1), (i_2, j_2) \dots (i_k, j_k) \dots (i_f, j_f)$

- Each path is associated with a cost $$D = \sum_{k=0}^{K-1} d(i_k, j_k)$$

  where K is the number of nodes across the path



➢ Search for the path with the optimal cost $D_{opt.}$

➢ The matching cost between template $\underline{r}$ and test pattern $\underline{t}$ is $D_{opt.}$

BELLMAN'S OPTIMALITLY PRINCIPLE

- Optimum path:

$$(i_0, j_0) \xrightarrow{opt} (i_f, j_f)$$

- Let (i,j) be an intermediate node, i.e.

$$(i_0, j_0) \rightarrow \ldots \rightarrow (i, j) \rightarrow \ldots \rightarrow (i_f, j_f)$$

Then write the optimal path through (i, j)

$$(i_0, j_0) \xrightarrow[(i,j)]{opt} (i_f, j_f)$$

❖ Bellman's Principle:

$$(i_0, j_0) \xrightarrow{opt} (i_f, j_f) = (i_0, j_0) \xrightarrow{opt} (i, j) \oplus (i, j) \xrightarrow{opt} (i_f, j_f)$$

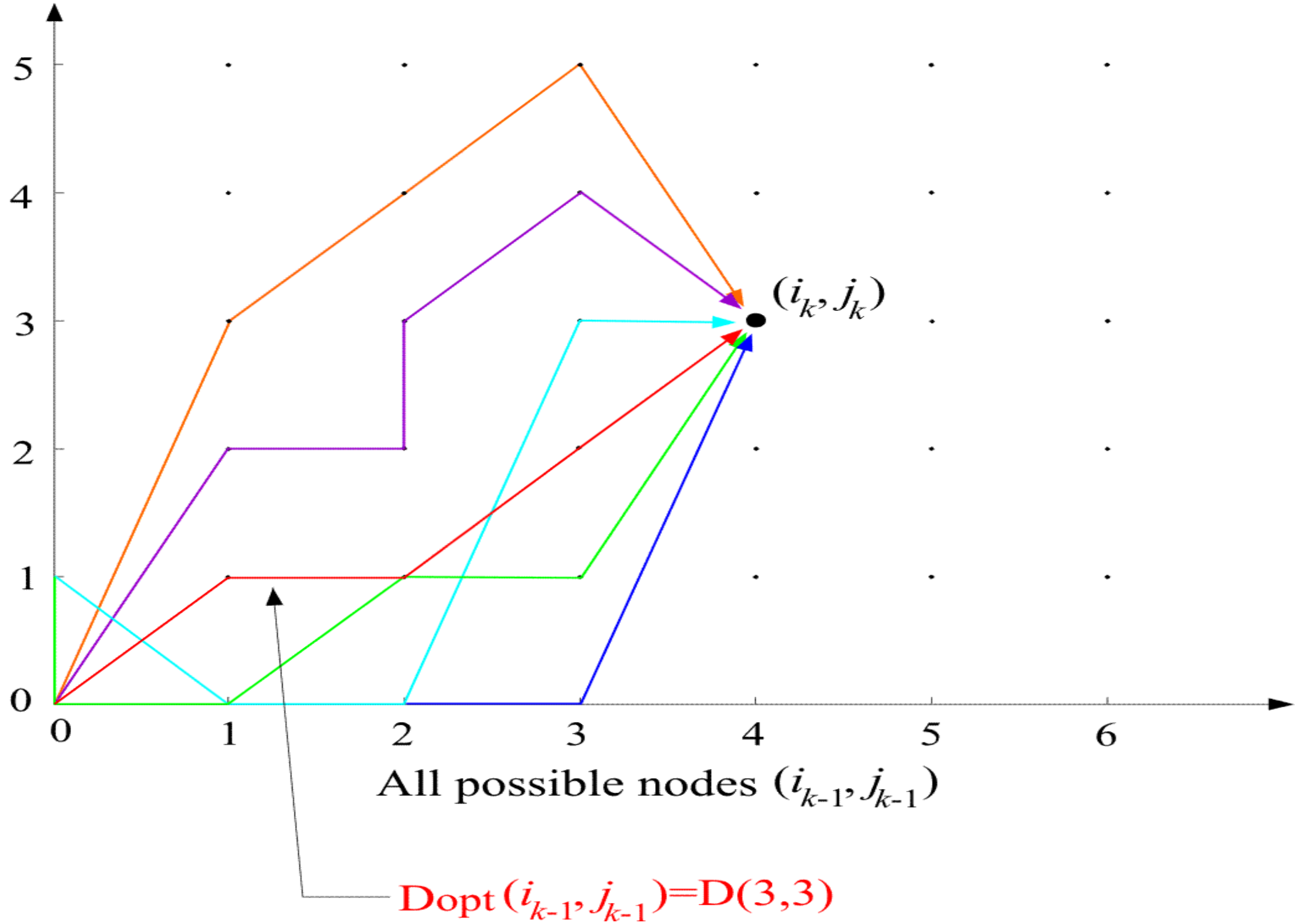
$\oplus$ denotes concatenation of paths

❖ In words: The overall optimal path from (i_0,j_0) to (i_f,j_f) through (i,j) is the concatenation of the optimal paths from (i_0,j_0) to (i,j) and from (i,j) to (i_f,j_f)

❖ Let $D_{min}(i,j)$ is the optimal path to reach (i,j) from (i_0,j_0), then Bellman's principle is stated as:

the overall minimum cost to reach node (i_k,j_k) is the minimum cost up to node (i_{k-1},j_{k-1}) plus the extra cost of the transition from (i_{k-1},j_{k-1}) to (i_k,j_k).

$$D_{\min}(i_k, j_k) = \min_{i_{k-1}, j_{k-1}} \{D_{\min}(i_{k-1}, j_{k-1}) + d(i_k, j_k \mid i_{k-1}, j_{k-1})\}$$



❖ The Edit distance (*Levenstein distance*)

- It is used for matching written words. Applications:
  - Automatic Editing
  - Text Retrieval

- The measure to be adopted for matching, must take into account:
  - Wrongly identified symbols
    e.g. “befuty” instead of “beauty”
  - Insertion errors, e.g. “bearuty”
  - Deletion errors, e.g. “beuty”

- The cost is based on the philosophy behind the so-called variational similarity, i.e.,
  - Measure the cost associated with converting one pattern to the other

- Edit distance: **Minimal** total number of changes, C, insertions I and deletions R, required to change pattern A into pattern B,

$$D(A,B) = \min_{j}[C(j) + I(j) + R(j)]$$

where j runs over All possible variations of symbols, in order to convert $A \longrightarrow B$

- Allowable predecessors and costs

  - Diagonal transitions $(i-1, j-1) \rightarrow (i, j)$

$$d(i, j \mid i-1, j-1) = \begin{cases} 0, & \text{if } t(i) = r(j) \\ 1, & t(i) \neq r(j) \end{cases}$$

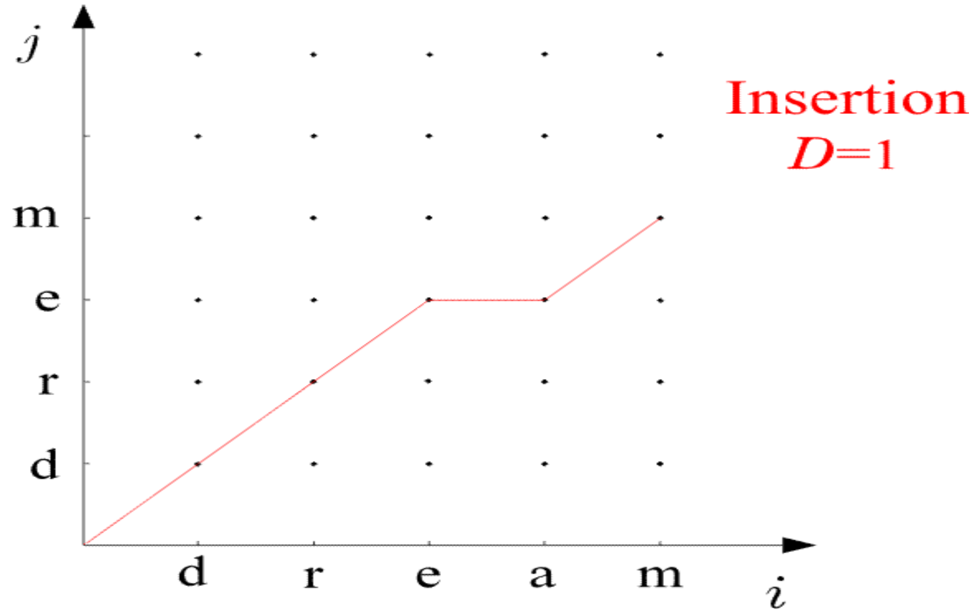
  - Horizontal Diagonal transitions

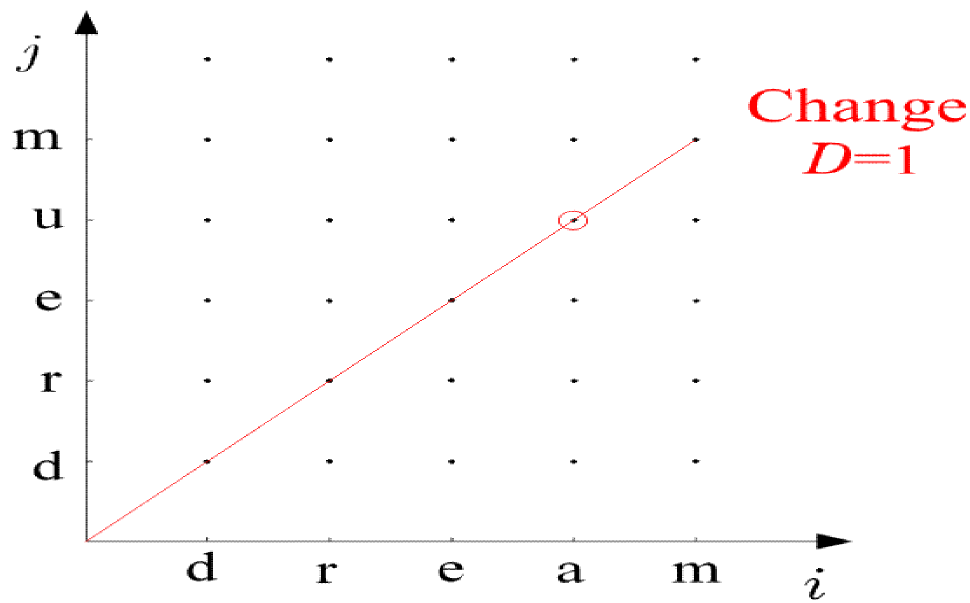
$$d(i, j | i-1, j) = 1$$

  - Vertical Diagonal transitions

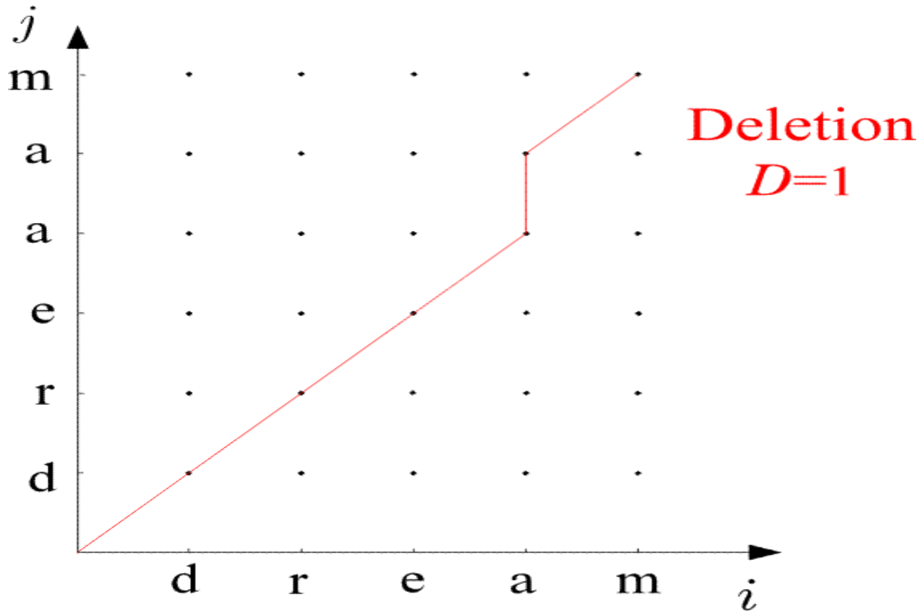
$$d(i, j | i, j-1) = 1$$

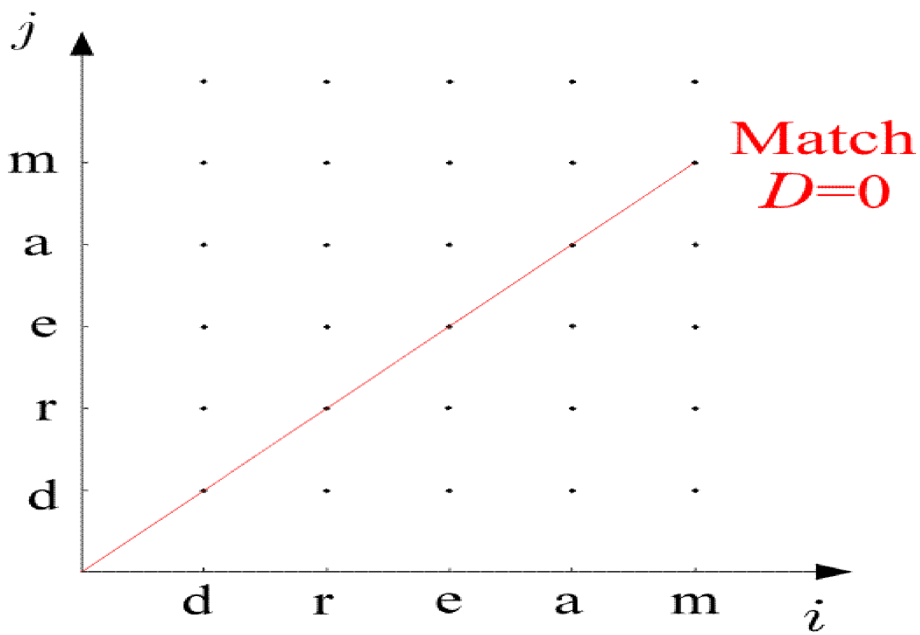
❖ Examples:





❖ Examples:





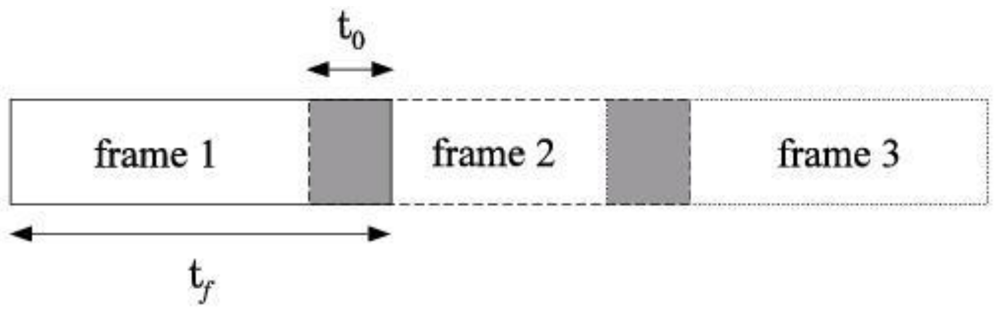
- The Algorithm
  - $D(0,0)=0$
  - For $i=1$, to I
    - $D(i,0)=D(i-1,0)+1$
  - END {FOR}
  - For $j=1$ to J
    - $D(0,j)=D(0,j-1)+1$
  - END{FOR}
  - For $i=1$ to I
    - For $j=1$, to J
      - $C_1=D(i-1,j-1)+d(i,j \mid i-1,j-1)$
      - $C_2=D(i-1,j)+1$
      - $C_3=D(i,j-1)+1$
      - $D(i,j)=\min(C_1,C_2,C_3)$
    - END {FOR}
  - END {FOR}
  - $D(A,B)=D(I,J)$

❖ Dynamic Time Warping in Speech Recognition

The isolated word recognition (IWR) will be discussed.

- The goal: Given a segment of speech corresponding to an unknown spoken word (test pattern), identify the word by comparing it against a number of known spoken words in a data base (reference patterns).

- The procedure:
  - Express the test and each of the reference patterns as sequences of feature vectors , $\underline{r}(i)$, $\underline{t}(j)$.
  - To this end, divide each of the speech segments in a number of successive frames.



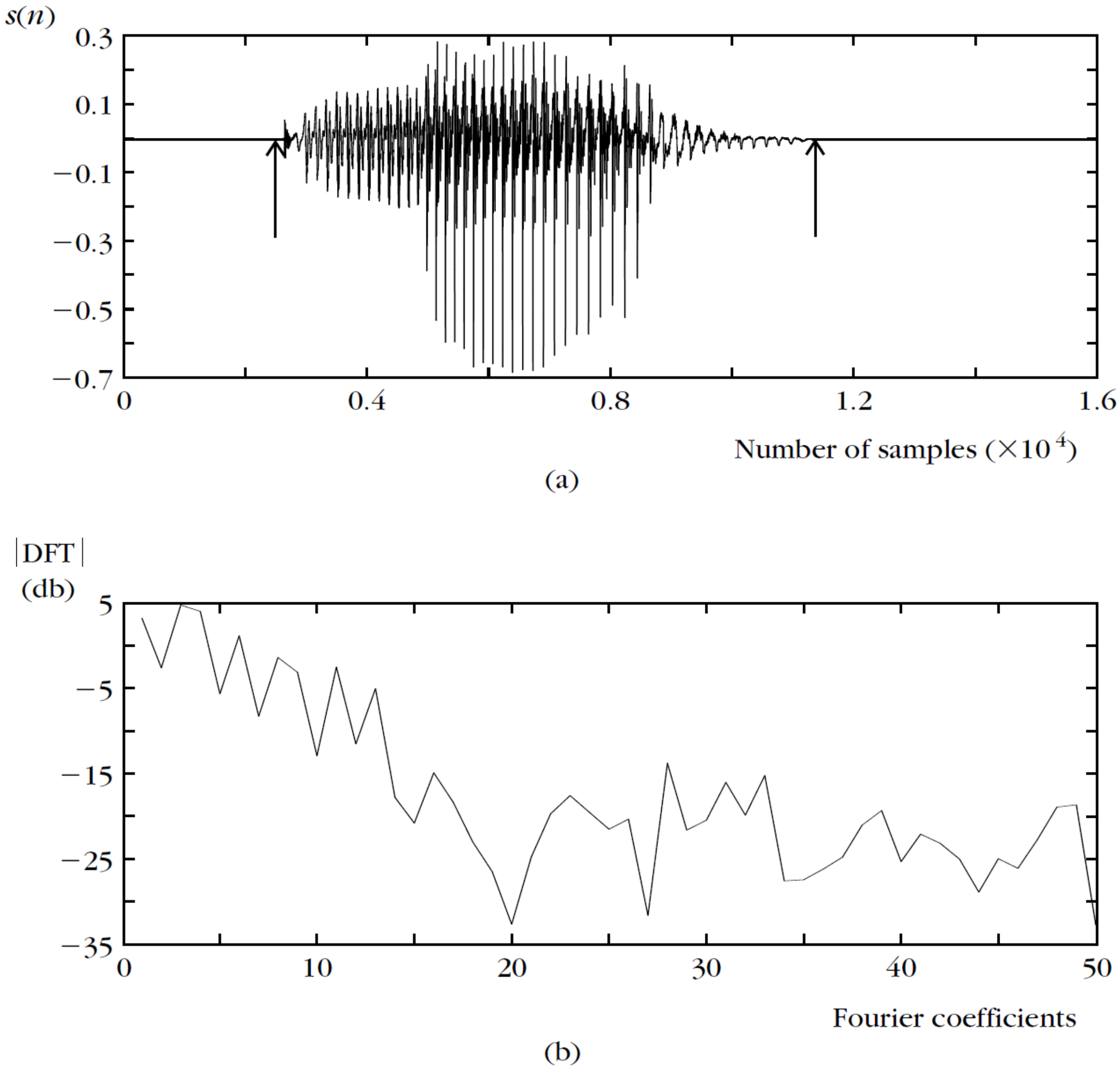
- For each frame compute a feature vector. For example, the DFT coefficients and use, say, ℓ of those:

$$\underline{r}(i) = \begin{bmatrix} x_i(0) \\ x_i(1) \\ \dots \\ \dots \\ x_i(\ell - 1) \end{bmatrix}, \; i = 1, \dots, I \qquad \underline{t}(j) = \begin{bmatrix} x_j(0) \\ x_j(1) \\ \dots \\ \dots \\ x_j(\ell - 1) \end{bmatrix}, \; j = 1, \dots, J$$

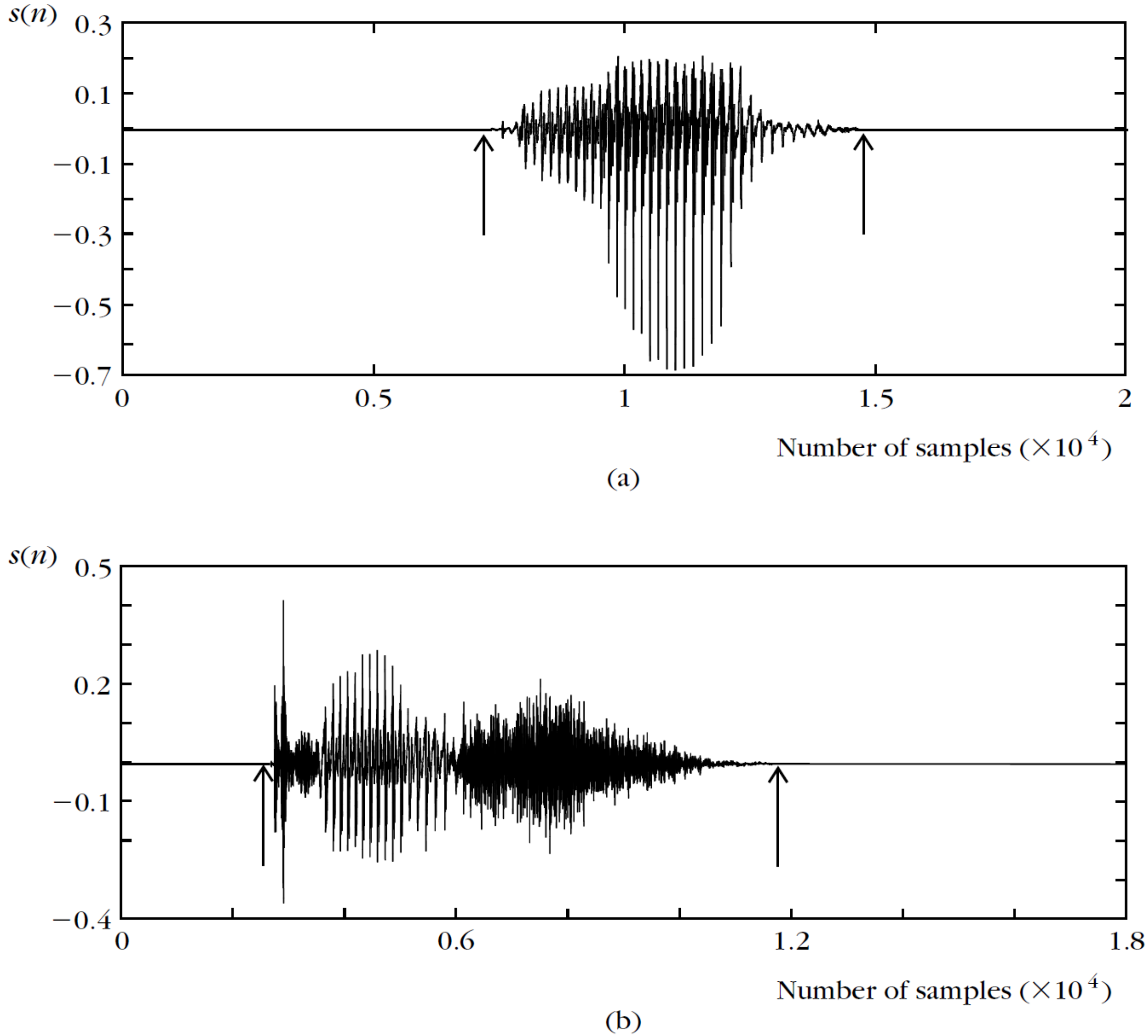
- Choose a cost function associated with each node across a path, e.g., the Euclidean distance

$$\left\| \underline{r}(i_k) - \underline{t}(j_k) \right\| = d(i_k, j_k)$$

- For each reference pattern compute the optimal path and the associated cost, against the test pattern.
- Match the test pattern to the reference pattern associated with the minimum cost.



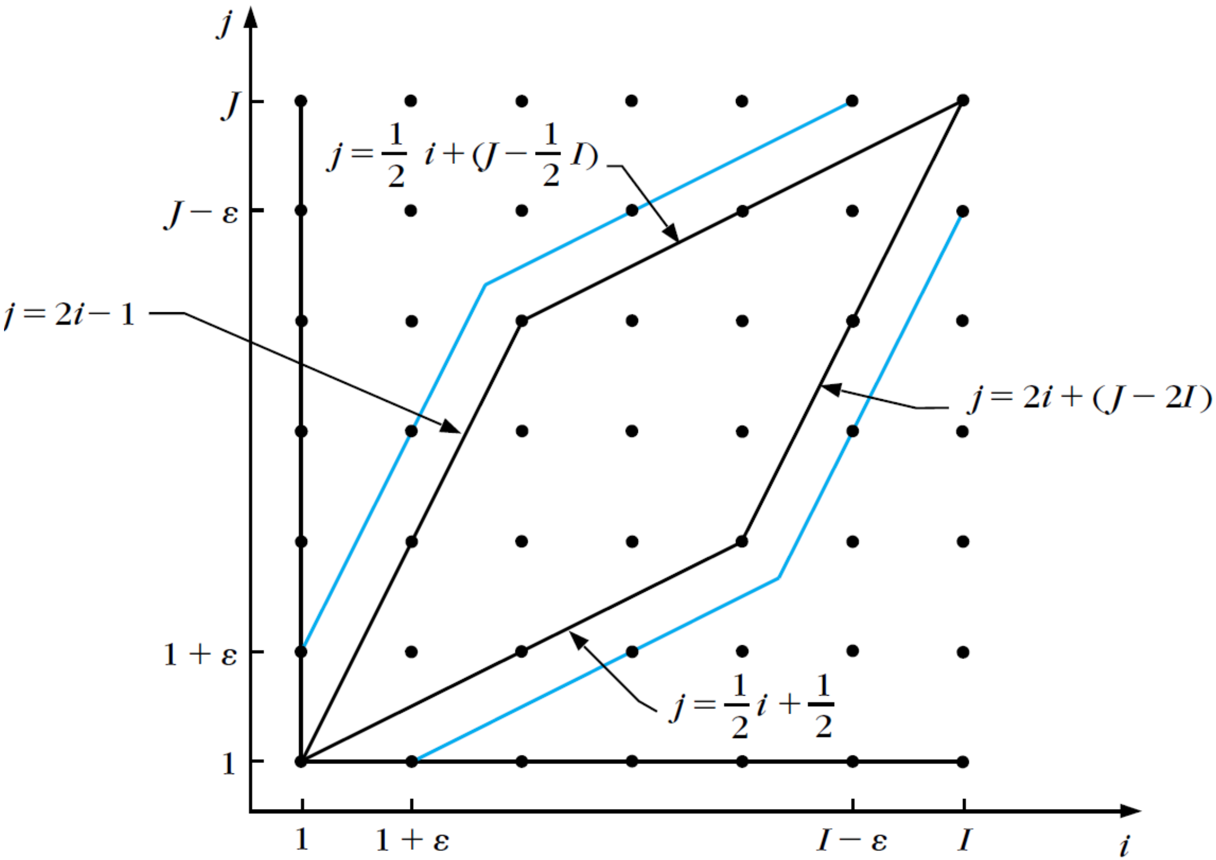
Plots of (a) the time sequence corresponding to the word "love" and (b) the magnitude of the DFT, in dB, for one of its frames.



Plots of the time sequences resulting from the words (a) “love” and (b) “kiss,” spoken by the same speaker

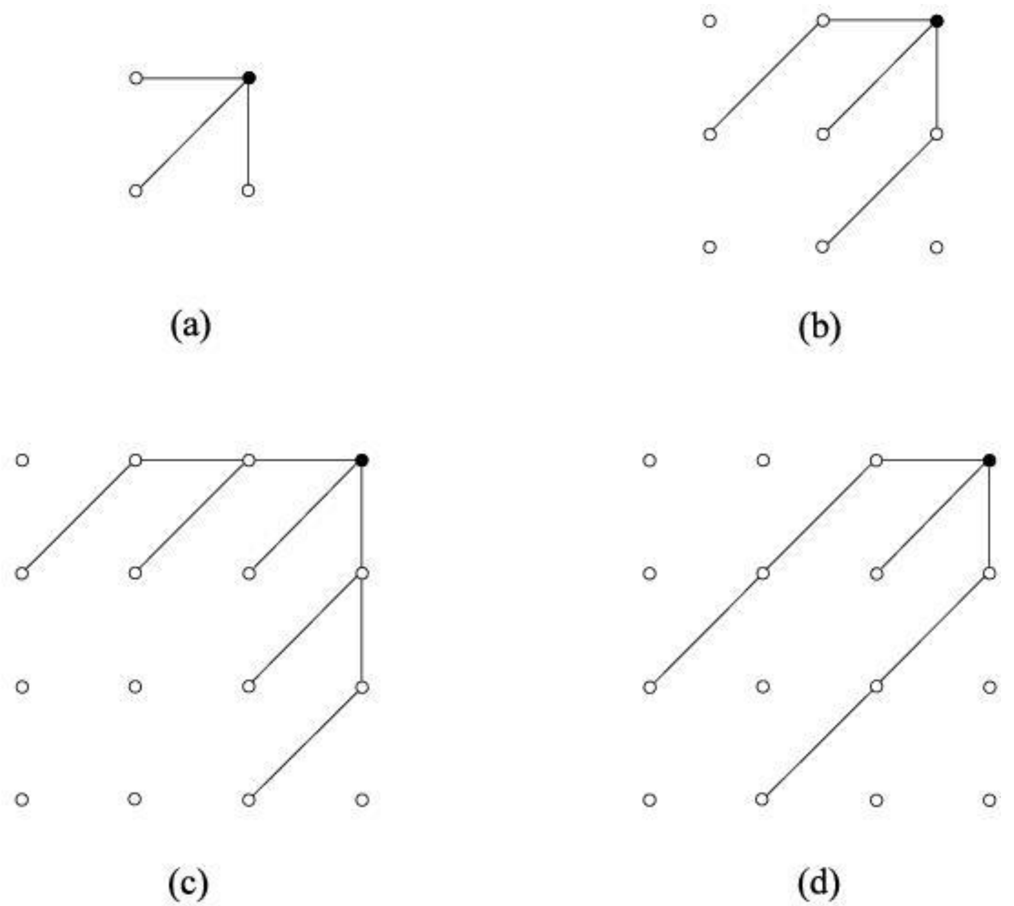
➢ Prior to performing the math one has to choose:

- The global constraints: Defining the region of space within which the search for the optimal path will be performed.



Itakura global constraints. The maximum compression/expansion factor is 2, and it determines the slope of the boundary line segments. The blue lines correspond to the same global constraints when the relaxed end-point constraints are adopted.

- The local constraints: Defining the type of transitions allowed between the nodes of the grid.



(a) there is no limit in the rate of expansion/compression (b) horizontal (vertical) transitions are allowed only after a diagonal transition (c) at most two successive horizontal (vertical) transitions are allowed only after a diagonal one.

تا پایان اسلاید حذف

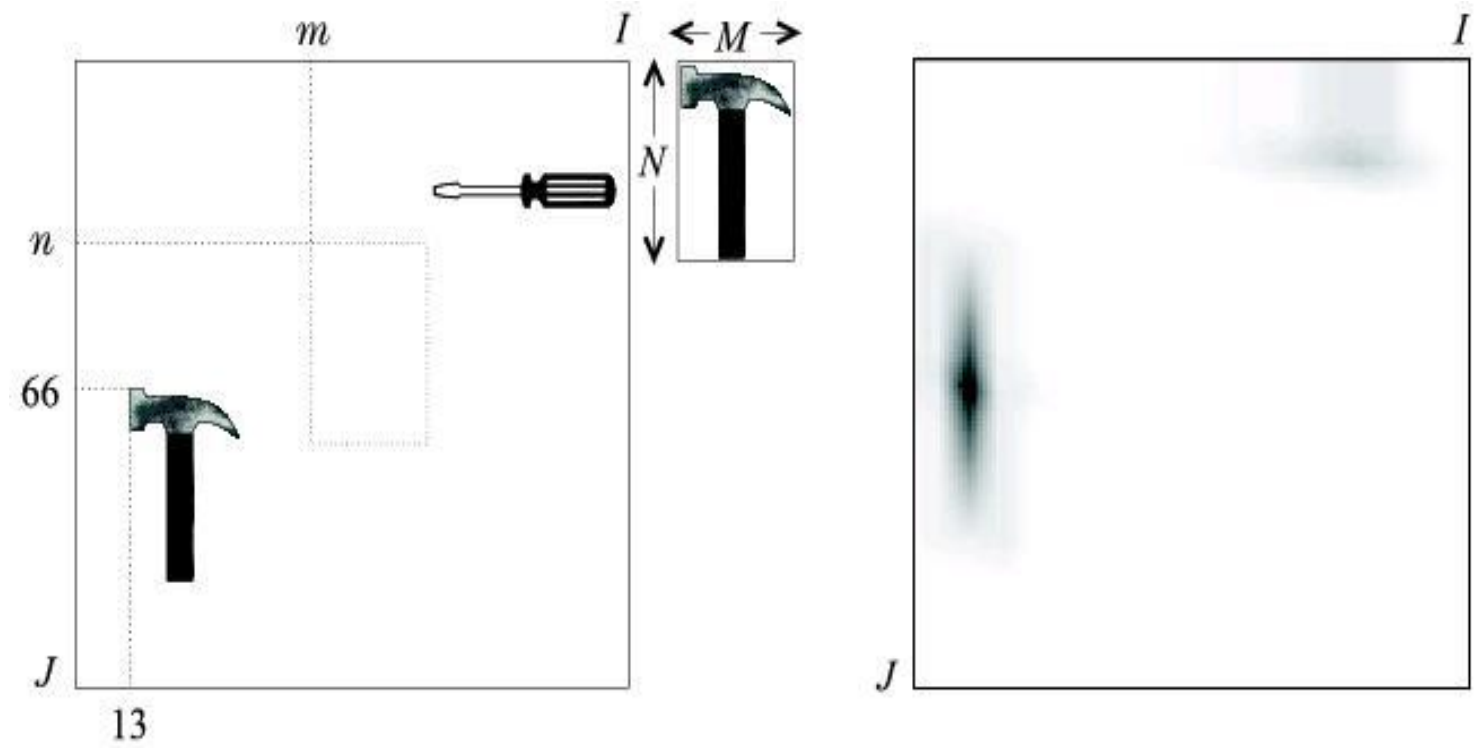
- Measures based on Correlations: The major task here is to find whether a specific known reference pattern resides within a given block of data. Such problems arise in problems such as target detection, robot vision, video coding. There are two basic steps in such a procedure:
  - Step 1: Move the reference pattern to all possible positions within the block of data. For each position, compute the "similarity" between the reference pattern and the respective part of the block of data.
  - Step 2: Compute the best matching value.

- Application to images: Given a reference image, $r(i,j)$ of MxN size, and an IxJ image array $t(i,j)$. Move $r(i,j)$ to all possible positions (m,n) within $t(i,j)$. Compute:
  - $$D(m,n) = \sum_i \sum_j \left| t(i,j) - r(i-m, j-n) \right|^2$$
    for every (m,n).
  - For all (m,n) compute the minimum.
  - The above is equivalent, for most practical cases, to compute the position (m,n) for which the correlation is maximum.
    - $$c(m,n) = \sum_i \sum_j t(i,j) \cdot r(i-m, j-n)$$
    - Equivalently, the normalized correlation can be computed as

$$c_N(m,n) = \frac{c(m,n)}{\sum_i \sum_j \left| t(i,j) \right|^2 \cdot \sum_i \sum_j \left| r(i,j) \right|^2}$$

– $c_N(m,n)$ is less than one and becomes equal to one only if

$$t(i, j) = \alpha \cdot r(i-m, j-n)$$

- 

(a) (b)

❖ Deformable Template Matching

In correlation matching, the reference pattern was assumed to reside within the test block of data. However, in most practical cases a **version** of the reference pattern lives within the test data, which is "similar" to the reference pattern, but **not exactly** the same. Such cases are encountered in applications such as content based retrieval from data bases.

- The philosophy: Given a reference pattern $r(i,j)$ known as prototype:
  - Deform the prototype to produce different variants. Deformation is described by the application of a parametric transform on $r(i,j)$:

$$T_{\xi}\left[r(i, j)\right]$$

- For different values of the parameter vector $\underline{\xi}$ the goodness of fit with the test pattern is given by the matching energy:

$$E_m\left(\underline{\xi}\right)$$

- However, the higher the deformation, the higher the deviation from the prototype. This is quantified by a cost known as deformation energy:

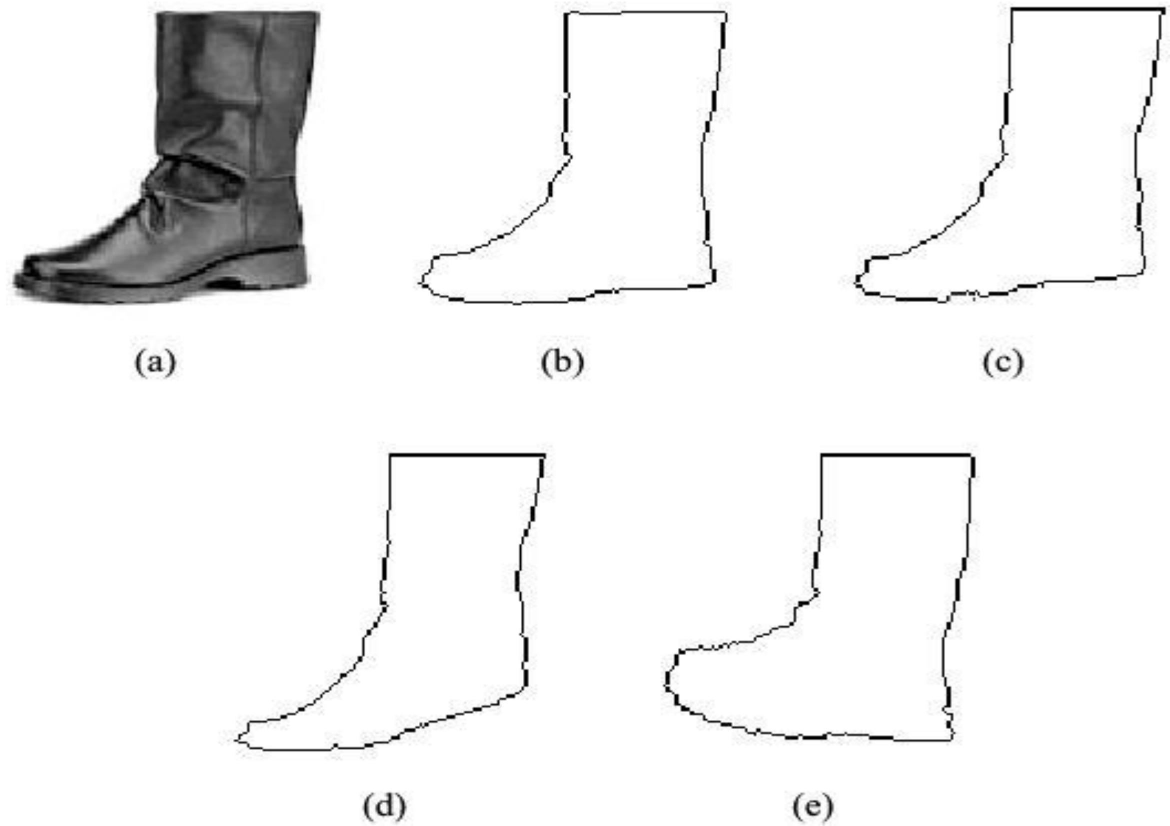
$$E_d\left(\underline{\xi}\right)$$

- In deformable template matching compute $\underline{\xi}$, so that

$$\underline{\xi} : \min_{\underline{\xi}} \left[E_m(\underline{\xi}) + E_d(\underline{\xi}) \right]$$

- Ideally, one should like to have both terms low: small deformation and small matching energy. This means that one can retrieve a pattern very similar to the prototype.

•



- Different choices of:
  - Transformation function
  - Matching Energy Cost
  - Deformation Energy cost

  are obviously possible.