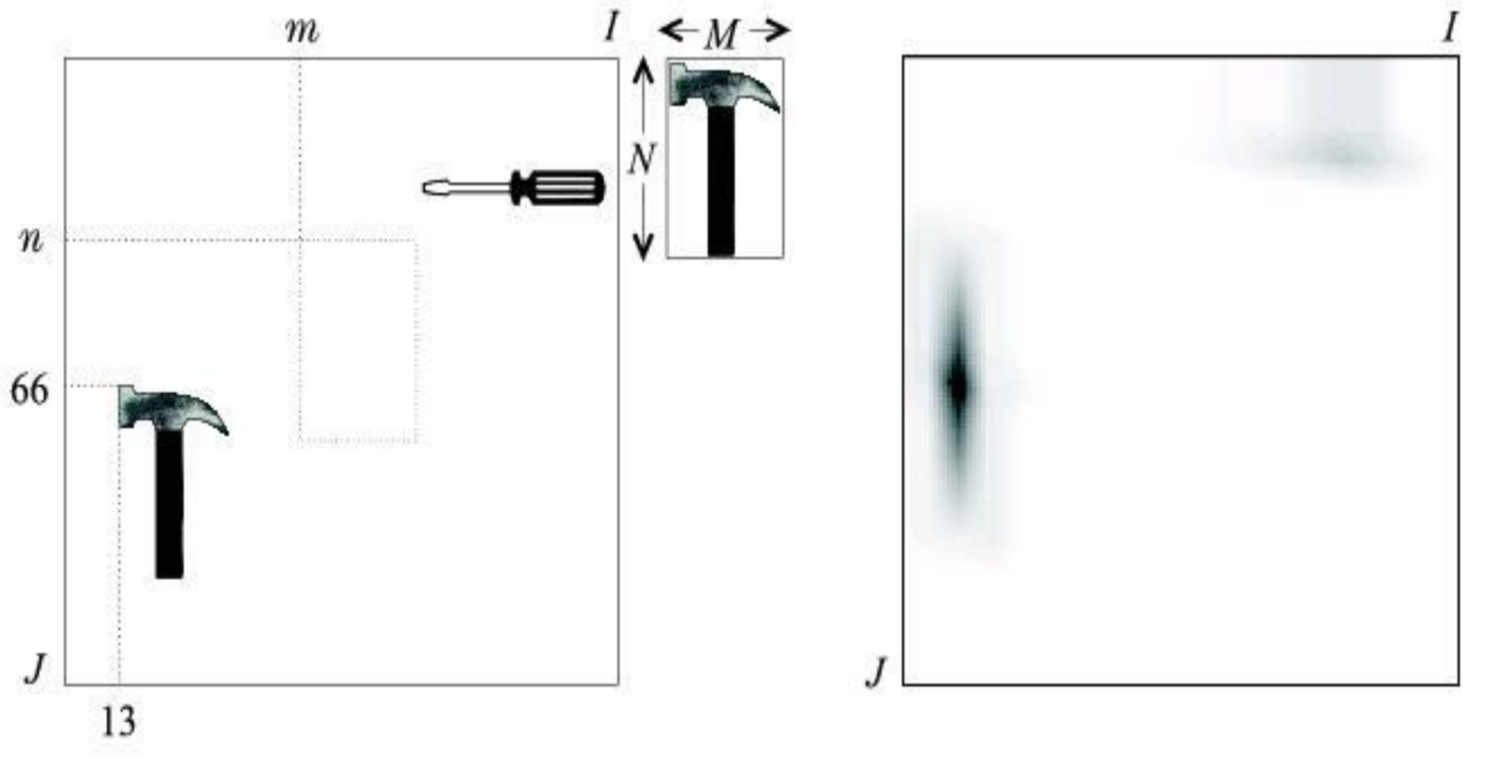
Template Matching



(a) (b)

Template Matching: Dynamic Time-Warping

- מדדים המבוססים על שיטות חיפוש של מסלול אופטימלי
התבניות בהן עוסקים הן מחרוזות של סמלים ידועים או וקטורי תכונות
  - כל תבנית מקור או מבחן מיוצגת על-ידי:
סדרה (מחרוזת) של פרמטרים מדודים $\left(i_0, j_0\right)$
  - לאיזה מרצפי או סדרות המקור תבנית המבחן הכי מתאימה?

Template Matching: Dynamic Time-Warping

נניח:

$$r(i), \quad i = 1, 2, \ldots, I$$

$$t(j), \quad j = 1, 2, \ldots, J$$

שתי סדרות של וקטורי תכונות של תבניות reference ו- test כלשהן.

באופן כללי:

$$I \neq J$$

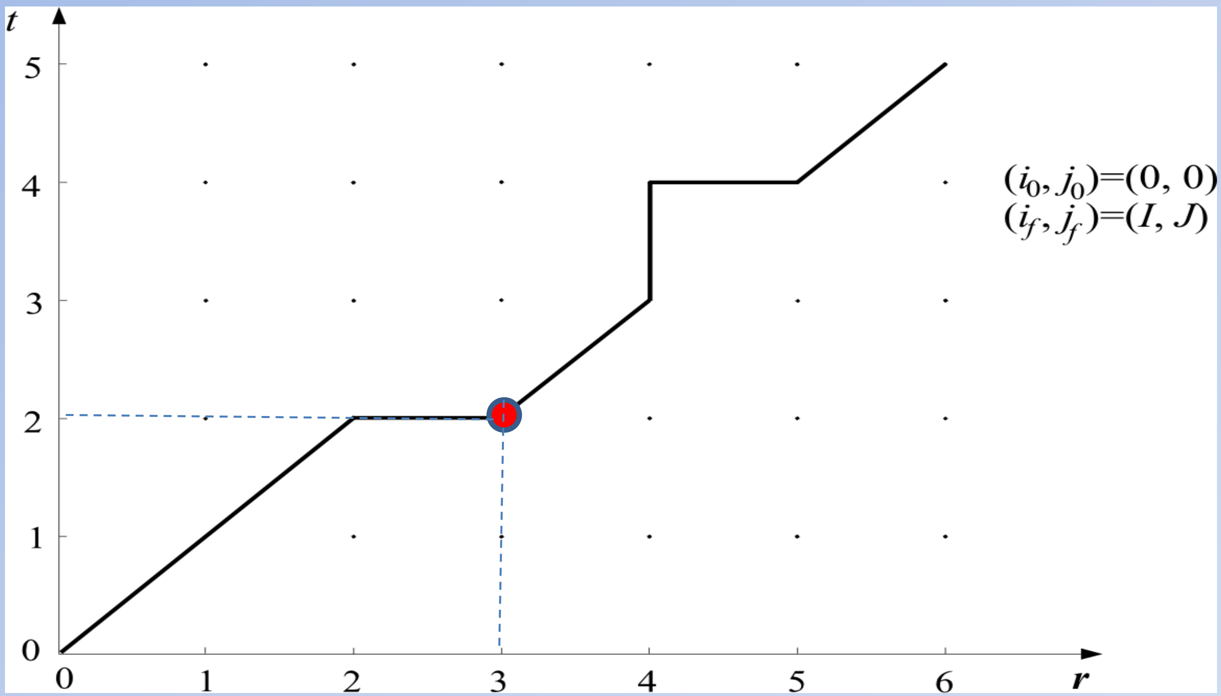
המטרה: לפתח מדד מרחק מתאים בין הסדרות.

לשם כך יוצרים סריג דו-מימדי.

Template Matching: Dynamic Time-Warping

לשם כך יוצרים סריג דו-מימדי.

$$r(i), \quad i = 1, 2, \ldots, I$$
$$t(j), \quad j = 1, 2, \ldots, J$$



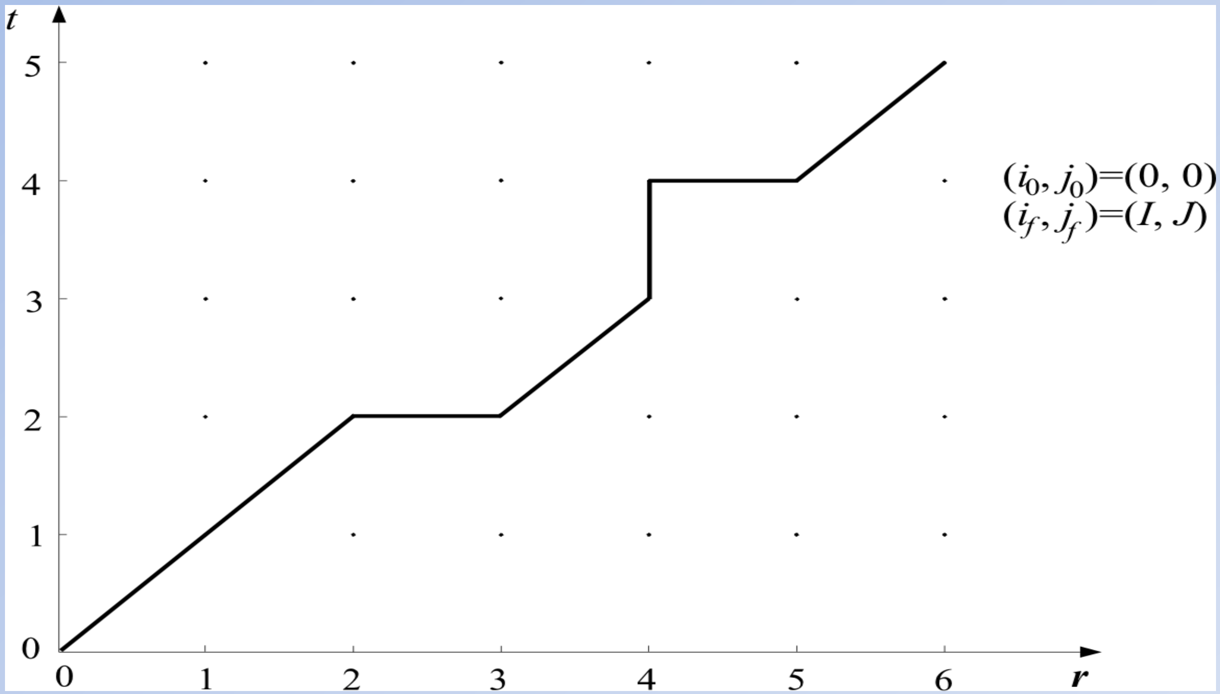
- כל קדקוד בסריג: התאמה בין האלמנטים של התבניות המתאימות.
  לדוגמא קדקוד (3,2) – מיפוי של r(3) ל- t(2).
- לכל קדקוד בסריג מתאים מרחק או מחיר.

Pattern Recognition ,Theodoridis and Koutroumbas, 2008

Template Matching: Dynamic Time-Warping

כל נקודה לאורך המסלול מסמנת את ההתאמה בין האלמנטים של תבנית המבחן ותבנית ה- reference.

$I = 5, J = 6$



סימון: $d(i, j)$

המרחק בין *r(i)* ל- *t(j)*

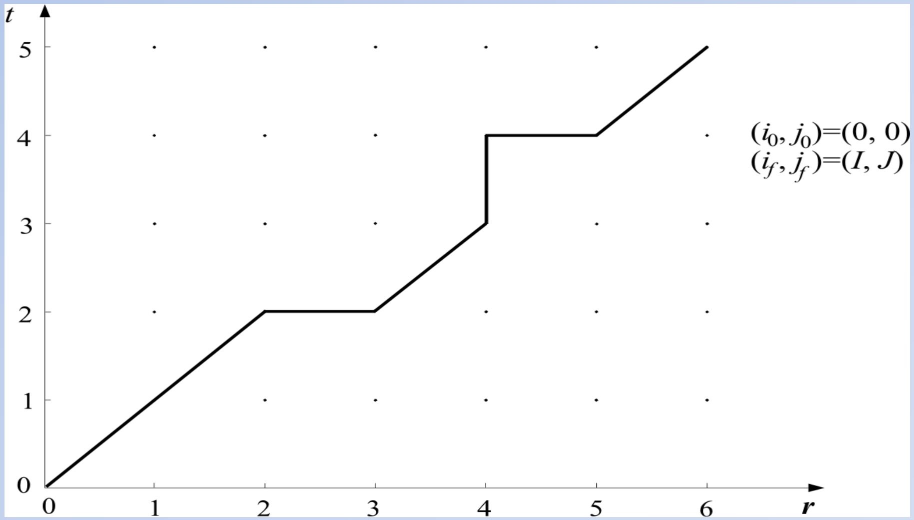
Pattern Recognition ,Theodoridis and Koutroumbas, 2008

Template Matchine: Dynamic Time-Warping

- 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

$$D = \sum_{k=0}^{K-1} d(i_k, j_k)$$

- where K is the number of nodes across the path (K=8 in the Figure).



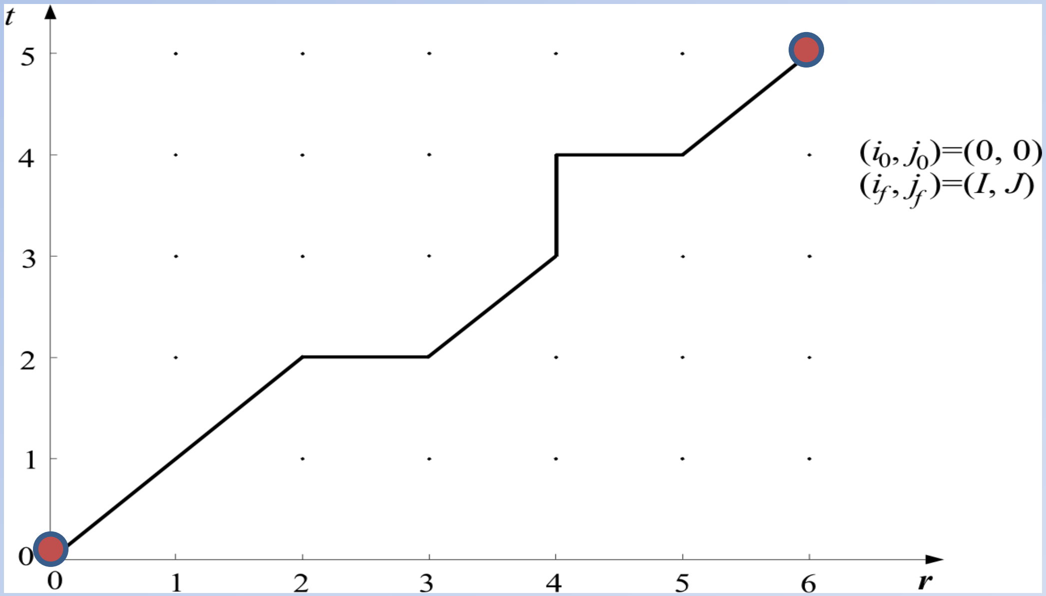
Pattern Recognition ,Theodoridis and Koutroumbas, 2008

Template Matchine: Dynamic Time-Warping

סימון: המחיר הכולל עד הקדקוד (i_k, j_k) : $D(i_k, j_k)$

קונבנציה: $D(0,0)=0, \quad d(0,0)=0,$

המסלול נקרא שלם אם $(i_0, j_0)=(0,0), \quad (i_F, j_F)=(I,J)$



Pattern Recognition ,Theodoridis and Koutroumbas, 2008

Dynamic Time-Warping

- המרחק בין שתי הסדרות – המינימום של D עבור כל המסלולים האפשריים.
- המסלול עם המחיר המינימלי – התאמה אופטימלית בין האלמנטים של שתי הסדרות (בדר"כ אורך שונה).

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

> The matching cost between template r and test pattern t is D_{opt}.

> The procedure of setting an optimal path – alignment or warping of the elements (test to reference string) according to the best matching score.

Dynamic Time-Warping

וריאציות:

- אילוץ משוכך של נקודת הסיום (relaxed end point constraints)

  המסלול לא חייב להיות שלם: לא חייב להתחיל ב- (i_0,j_0) ולהסתיים ב- (i_f, j_f)

- עלות דיפרנציאלית - עלות מעבר מסוים עשויה להיות שונה מזו של מעבר אחר.

  לדוגמא – המחיר של הקדקוד (i_k, j_k) תלוי מאין מגיעים אליו (i_{k-1}, j_{k-1})

  העלות d היא אם-כן מהצורה: $d(i_k, j_k|i_{k-1}, j_{k-1})$

$$D = \sum_{k=0}^{K-1} d(i_k, j_k \mid i_{k-1}, j_{k-1})$$

- מכפלה במקום סכום:

$$D = \prod_{k=0}^{K-1} d(i_k, j_k \mid i_{k-1}, j_{k-1})$$

- חיפוש מסלול עלות מקסימלית (במקום מינימלית)

- לכל המקרים תנאי התחלה מתאימים

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)

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

Pattern Recognition ,Theodoridis and Koutroumbas, 2008

Optimal path finding

- כיצד נבחר את המסלול האופטימלי?
- חישוב כל המסלולים האפשריים – יקר חישובית
- אלגוריתם תכנות דינמי להפחתת סיבוכיות חישובית

שימוש בעיקרון האופטימליות של Bellman

Pattern Recognition ,Theodoridis and Koutroumbas, 2008

BELLMAN'S OPTIMALITLY PRINCIPLE

- Bellman's Principle:

נסמן: המסלול האופטימלי מ- (i_0,j_0) ל- (i_f,j_f)

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

נניח כי (i_k,j_k) צומת ביניים, ונסמן את המסלול האופטימלי עם האילוץ לעבור דרך (i_k,j_k)

$$(i_0, j_0) \xrightarrow[(i_k, j_k)]{opt} (i_f, j_f)$$

BELLMAN'S OPTIMALITLY PRINCIPLE

- Bellman's Principle:

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

$$(i_0, j_0) \xrightarrow{opt} (i_k, j_k) \oplus (i_k, j_k) \xrightarrow{opt} (i_f, j_f) \multimap$$

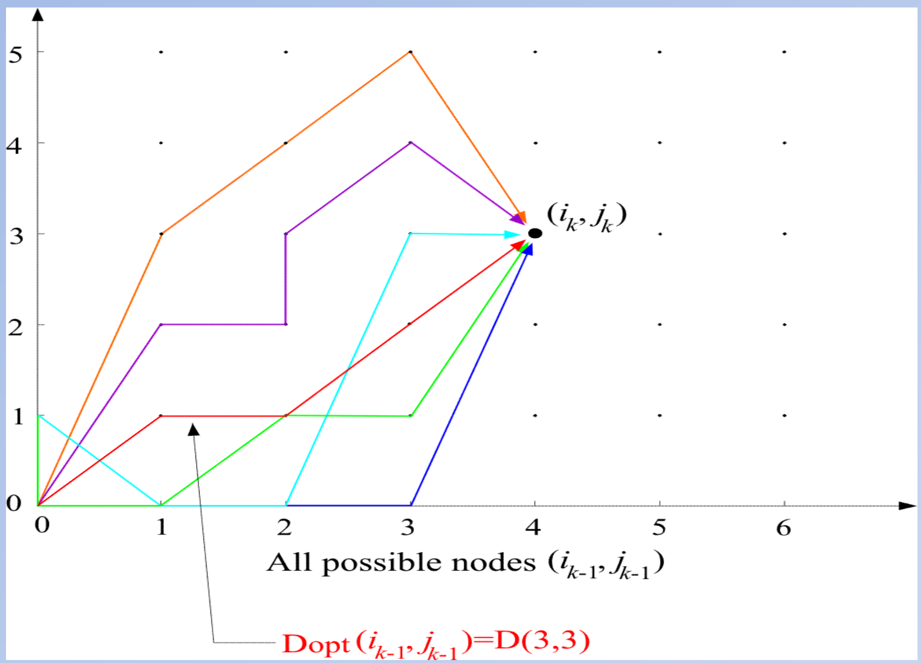
- In words: The overall optimal path from (i_0,j_0) to (i_f,j_f) through (i,j) is the concatenation of the optimal paths from (i_0,j_0) to (i_k,j_k) and from (i_k,j_k) to (i_f,j_f)

- ההשלכות של העיקרון: אם הגענו לצומת ה- (i_k,j_k) דרך המסלול האופטימלי עד לנקודה זו, כדי להגיע ל- (i_f,j_f) באופן אופטימלי, צריך למצוא את המסלול האופטימלי מ- (i_k,j_k) ל-(i_f,j_f)

Pattern Recognition ,Theodoridis and Koutroumbas, 2008

Dynamic Time-Warping

המסלול האופטימלי ל- (i_k,j_k) מ- (i_0,j_0) עובר דרך הנקודה (i_{k-1},j_{k-1})



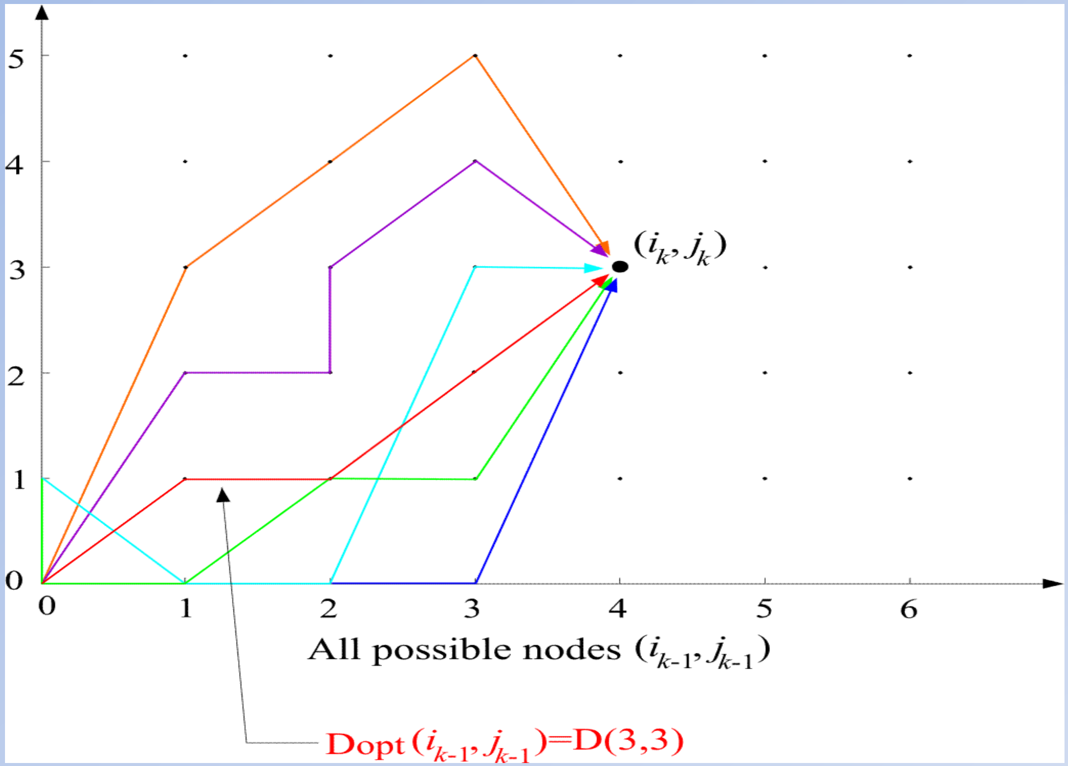
בציור: לכל צומת לאורך המסלול יש מעבר רק מקבוצה של צמתים מותרים קודמים, המגדירים את ה- local constraints

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

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

Dynamic Time-Warping

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



במקום opt אפשר לרשום min או max

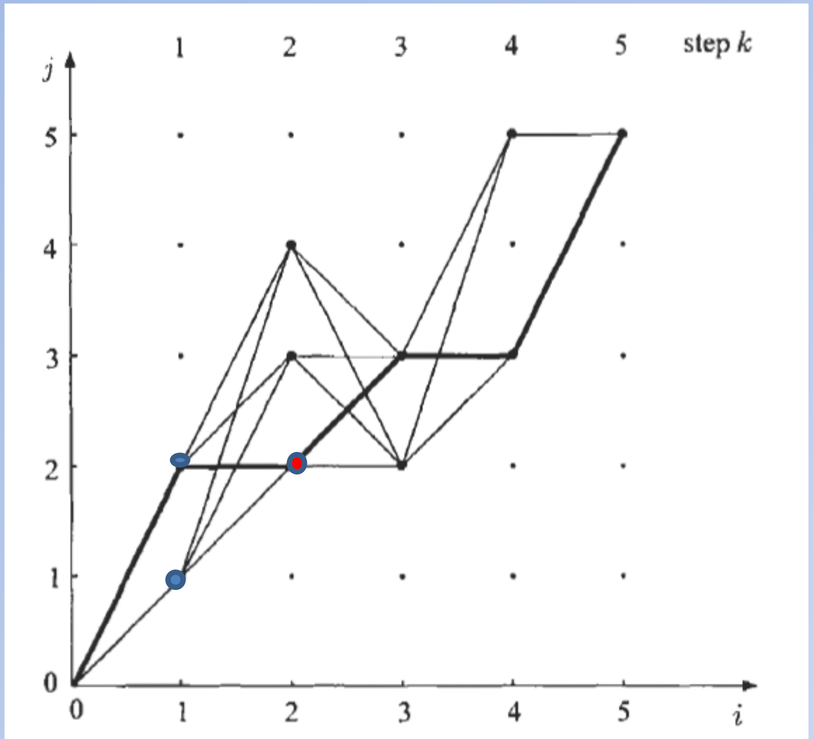
לסיכום:
המחיר האופטימלי (המינימלי) להגיע ל- (i_k,j_k) הוא המחיר האופטימלי של הסכום: המחיר האופטימלי עד לנקודה (i_{k-1},j_{k-1}) ועוד מחיר המעבר מ- (i_{k-1},j_{k-1}) ל-(i_k,j_k).

בנוסף, החיפוש מוגבל רק לנקודות המותרות ל- (i_k,j_k) - אילוצים מקומיים.
הפרוצדורה מופעלת על כל נקודות הסריג, או לפי אילוצים גלובליים.

Dynamic Time-Warping

דוגמא: נדגים את התאמת התבניות, ונראה איך משתמשים במשוואה הרקורסיבית (min).

נגדיר:
$$D(0,0)=0$$



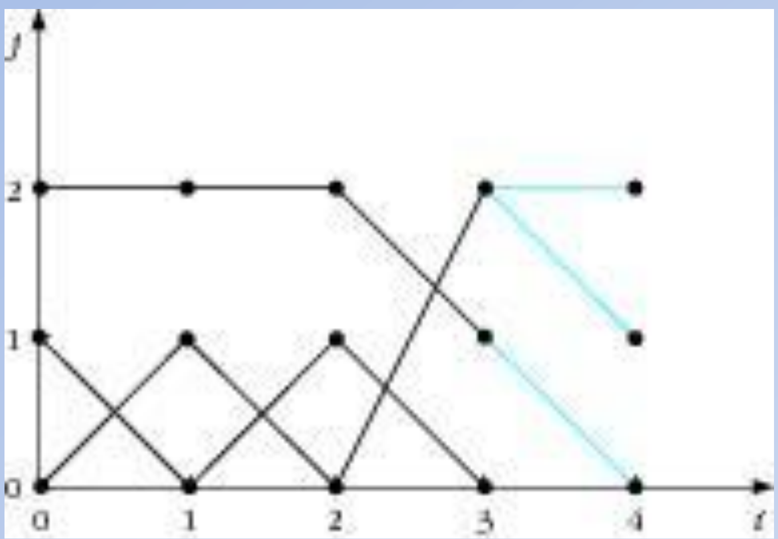
- המחירים של D(i,j) מחושבים עבור כל הנקודות המותרות (1,2) ,(1,1), לפי הנוסחה הרקורסיבית.
- בהמשך מחשבים את המחירים של $D(i_2,j_2)$ עבור k=2
- לנקודה (2,2)אפשר להגיע לדוגמא מ- (1,1), או מ- (1,2)

המסלול האופטימלי לנקודה (2,2) :

$$D\min(i_2,j_2)=D_{\min}(2,2)=$$
$$\min_{(i_1,j_1)}\{D_{\min}(i_{k-1},j_{k-1})+d(i_k,j_k\,|\,i_{k-1},j_{k-1})\}=$$
$$\min_{(i_1,j_1)}\{D(1,1)+d\big((2,2)\,|\,(1,1)\big),D(1,2)+d\big((2,2)\,|\,(1,2)\big)\}$$

Dynamic Time-Warping: example

- בציור: המסלולים האופטימליים מצעד k=0 עד צעד k=3.
  הסריג כולל רק 3 צמתים לכל צעד.
- מטרת התרגיל: לקבוע מהו המסלול האופטימלי לצעד k=4 בהינתן המסלולים האופטימליים עד לצעד k=3. הפעילו את עיקרון האופטימליות של Bellman.
- נתונים:



$$D_{\min}(3,0)=0.8,$$
$$D_{\min}(3,1)=1.2,$$
$$D_{\min}(3,2)=1.0$$
$$d\big((4,j_4)\,|\,(3,j_3)\big),\quad j_3=0,1,2\quad j_4=0,1,2$$

Nodes	$(4,0)$	$(4,1)$	$(4,2)$
$(3,0)$	0.8	0.6	0.8
$(3,1)$	0.2	0.3	0.2
$(3,2)$	0.7	0.2	0.3

לדוגמא: עלות המעבר מהצומת 3,1 לצומת 4,2 היא 0.2

Dynamic Time-Warping: example

$$\text{Total cost from } (3,0) \; to \; (4,0): 0.8 + 0.8 = 1.6$$

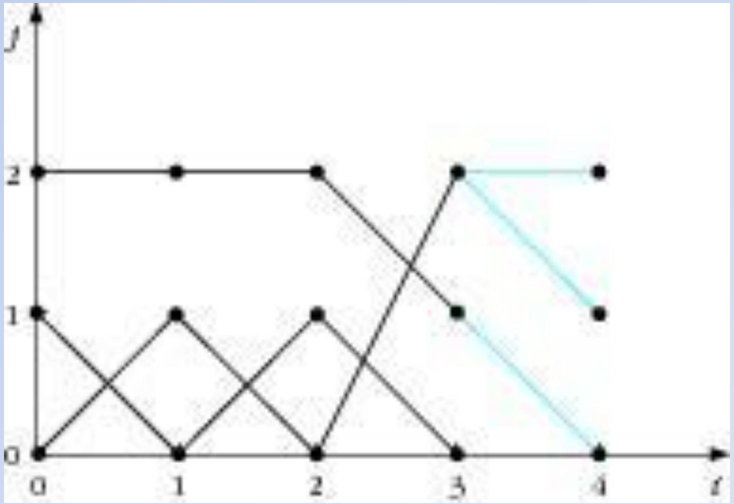
$$\text{Total cost from } (3,1) \; to \; (4,0): 1.2 + 0.2 = 1.4$$

$$\text{Total cost from } (3,2) \; to \; (4,0): 1.0 + 0.7 = 1.7$$

$D_{\min}(3,0) = 0.8,$
$D_{\min}(3,1) = 1.2,$
$D_{\min}(3,2) = 1.0$
$d\left((4, j_4) | (3, j_3)\right), \quad j_3 = 0,1,2 \quad j_4 = 0,1,2$

לכן המסלול המקסימלי המצטבר ל- (4,0) מגיע מ- (3,1).
חשבו את העלות המצטברת לצמתים (4,1) ו- (4,2) והסבירו את הציור.

$Nodes$	$(4,0)$	$(4,1)$	$(4,2)$
$(3,0)$	0.8	0.6	0.8
$(3,1)$	0.2	0.3	0.2
$(3,2)$	0.7	0.2	0.3



Dynamic Time-Warping: solution

$D_{\min}(3,0) = 0.8,$
$D_{\min}(3,1) = 1.2,$
$D_{\min}(3,2) = 1.0$
$d\left(\left(4, j_4\right) | \left(3, j_3\right)\right),$
$j_3 = 0,1,2 \quad j_4 = 0,1,2$

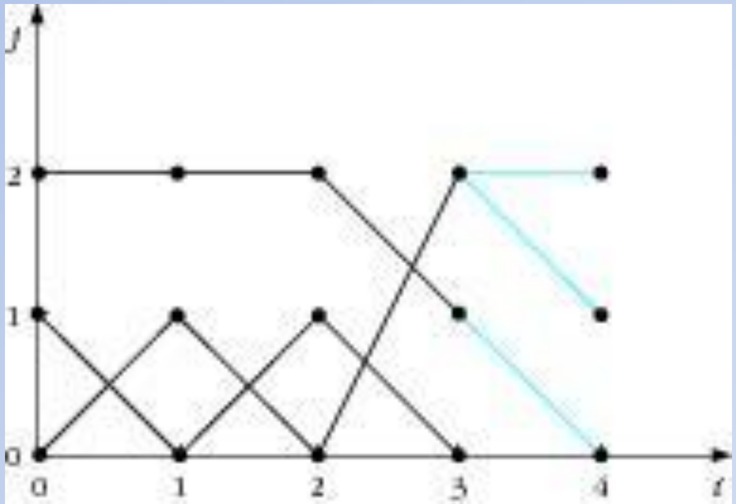
$$Total\,\cos t\ \ from\left(3,0\right) to\left(4,1\right): 0.6 + 0.8 = 1.4$$

$$Total\,\cos t\ \ from\left(3,1\right) to\left(4,1\right): 1.2 + 0.3 = 1.5$$

$$Total\,\cos t\ \ from\left(3,2\right) to\left(4,1\right): 1.0 + 0.2 = 1.2$$

Nodes	$(4,0)$	$(4,1)$	$(4,2)$
$(3,0)$	0.8	0.6	0.8
$(3,1)$	0.2	0.3	0.2
$(3,2)$	0.7	0.2	0.3

לכן המסלול המקסימלי המצטבר ל- (4,1) מגיע מ- (3,2).



Dynamic Time-Warping

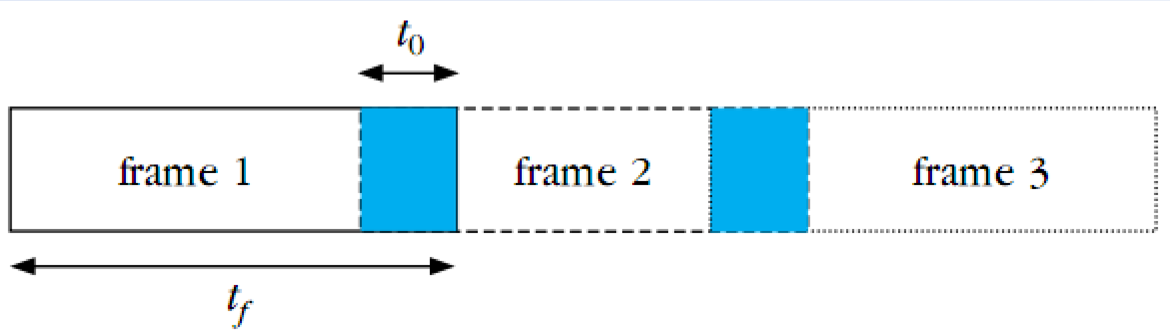
- Dynamic Time Warping in Speech Recognition

  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.



Pattern Recognition ,Theodoridis and Koutroumbas, 2008

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

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

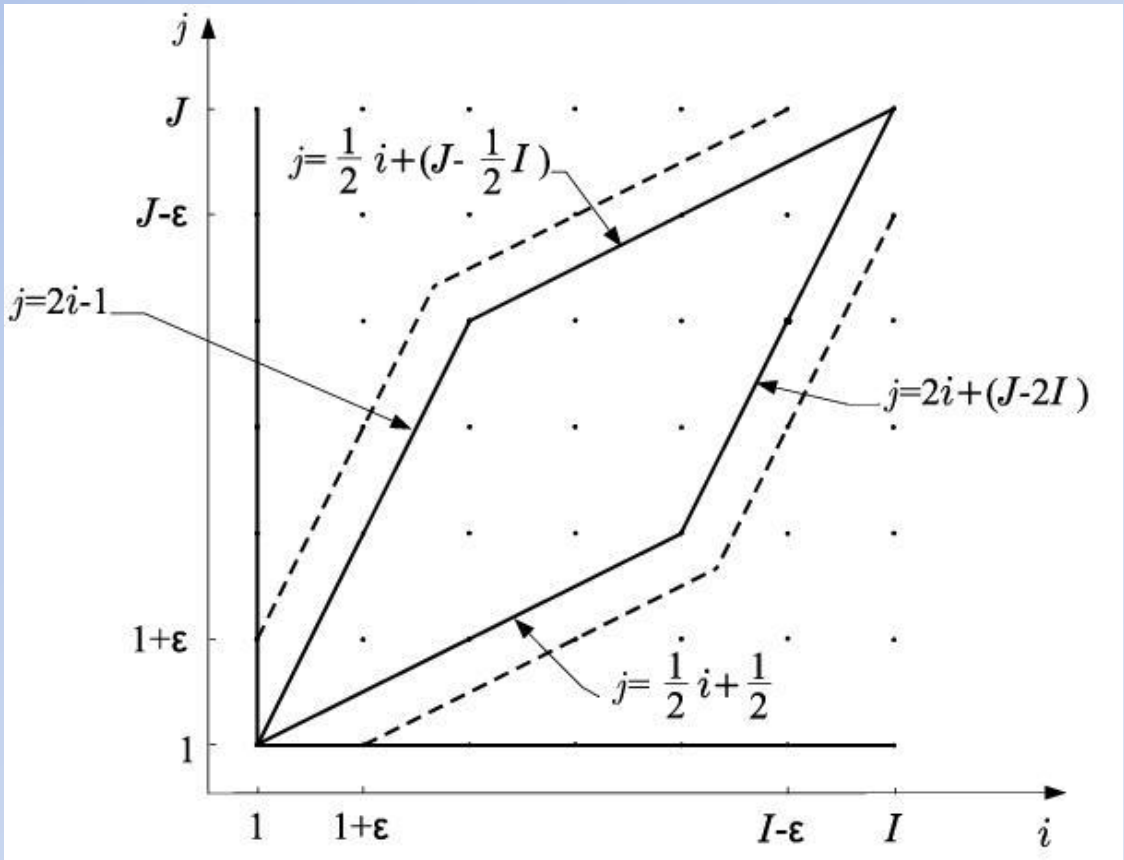
- Choose a cost function associated with each node across a path, e.g., the Euclidean distance

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

- 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.

Dynamic Time-Warping

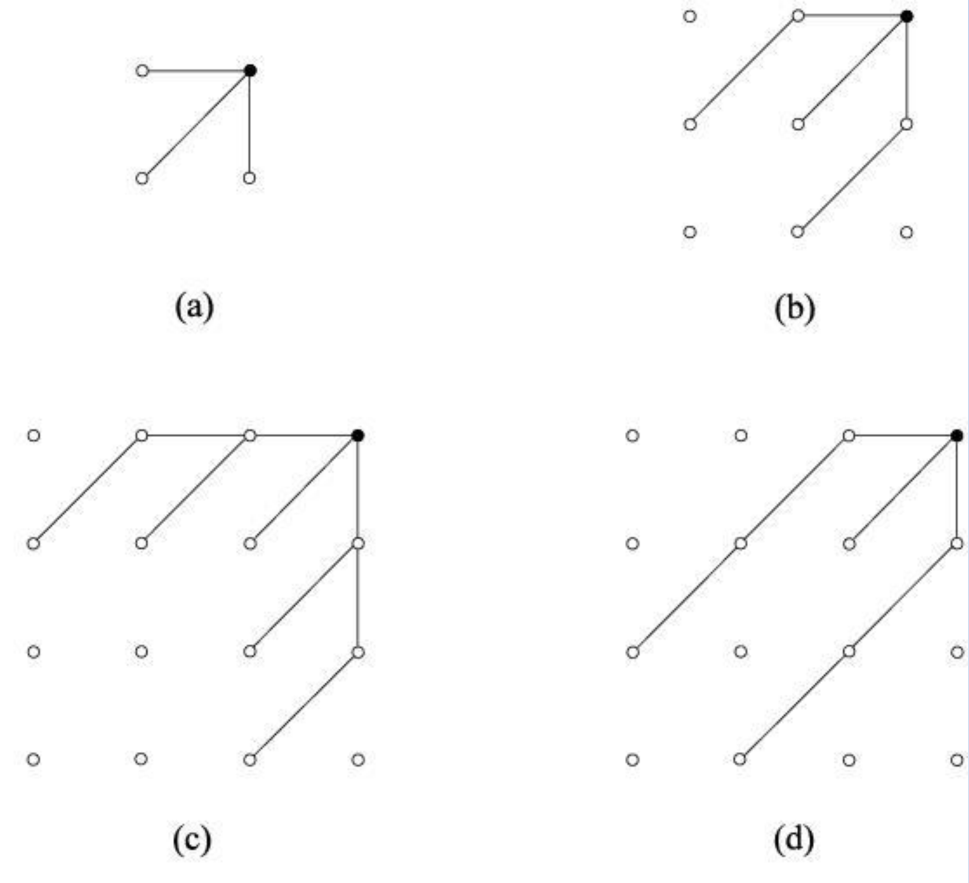
– Prior to performing the math one has to choose:
  - The global constraints: Defining the region of space within which the search for the optimal path will be performed.



Pattern Recognition ,Theodoridis and Koutroumbas, 2008

Dynamic Time-Warping

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



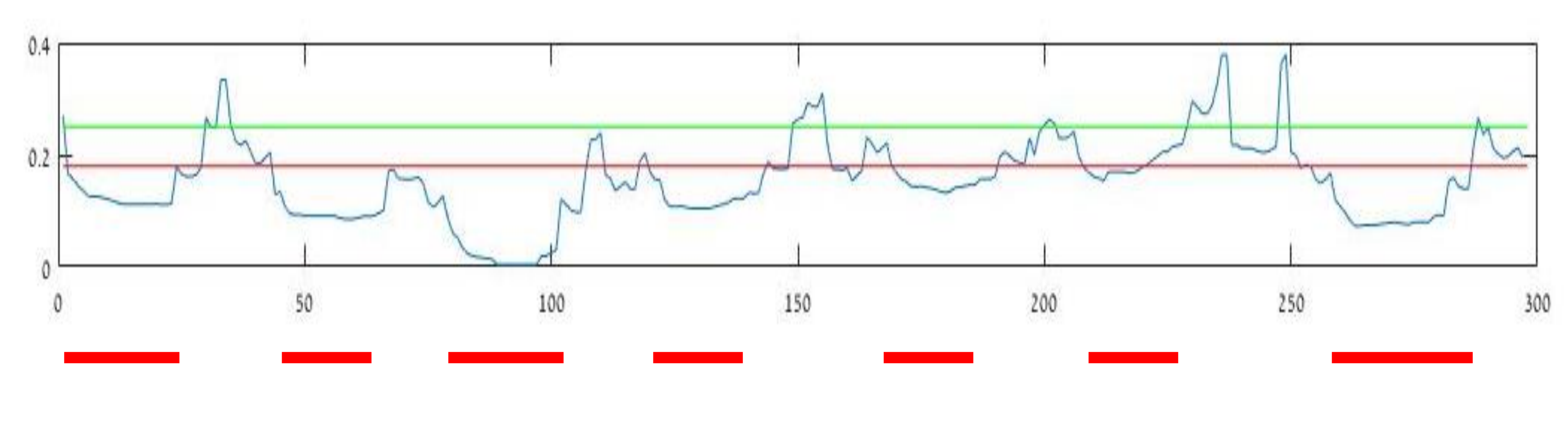
Pattern Recognition ,Theodoridis and Koutroumbas, 2008

Distance Measure Computation

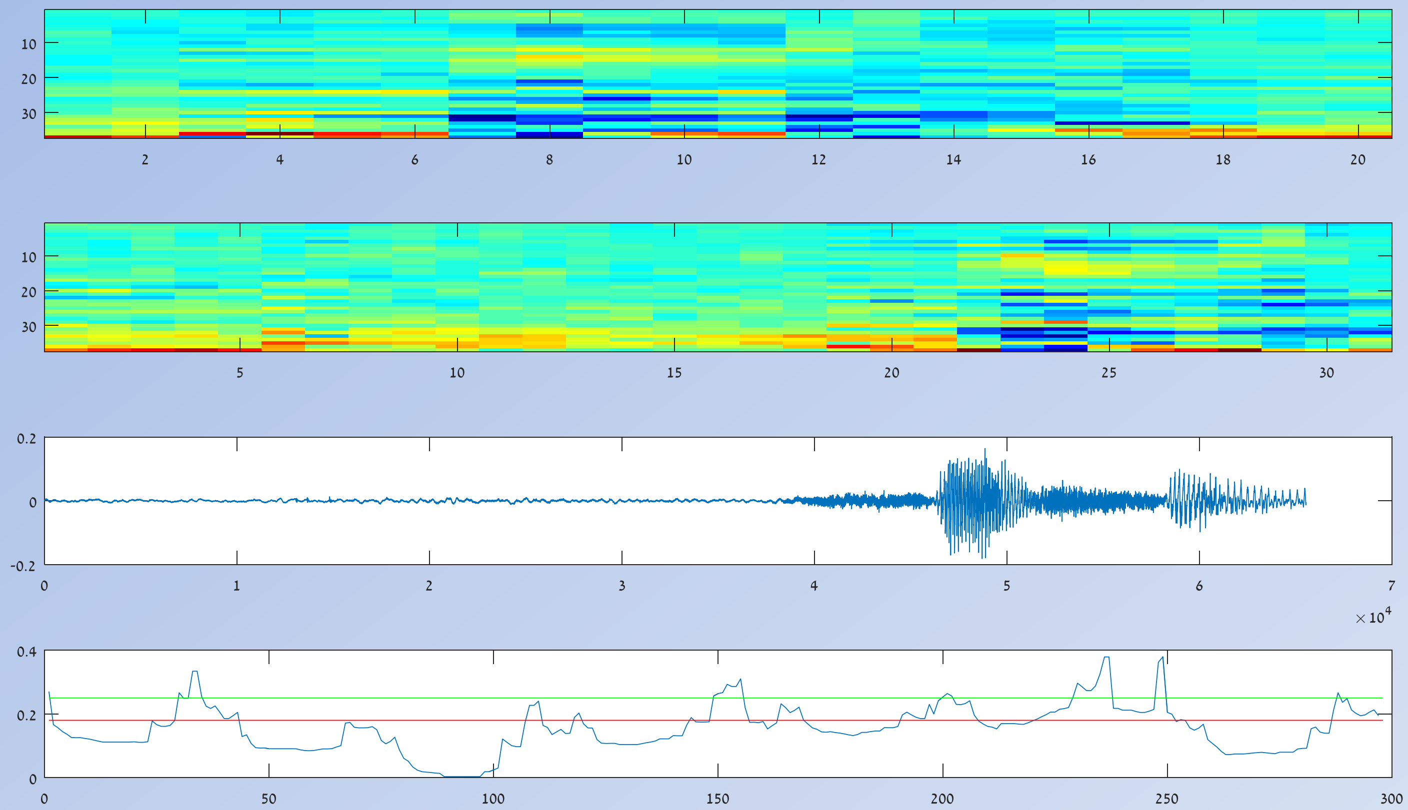
המרחק בין התבנית (template) ומטריצת הקלט (input matrix) הוא סכום המרחקים לאורך המסלול האופטימלי, עם מיצוע לפי אורך המסלול, כאשר מגדירים את המרחק בין כל עמודת תבנית ובין כל עמודה של מטריצת הקלט על-ידי:

$$dist\left(t_i, r_j\right) = \frac{1}{2}\left(1 - \frac{\left\langle t_i, r_j \right\rangle}{\left\| t_i \right\| \cdot \left\| r_j \right\|}\right)$$

Threshold Setting



Double-check Mechanism



References

- Pattern Recognition ,Theodoridis and Koutroumbas, 2008