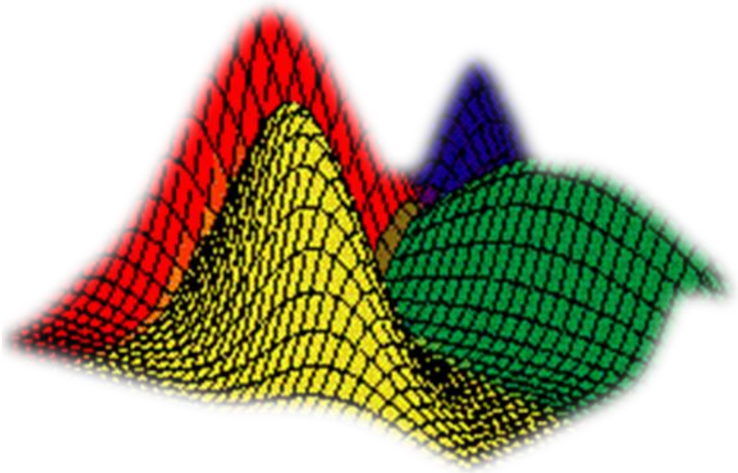
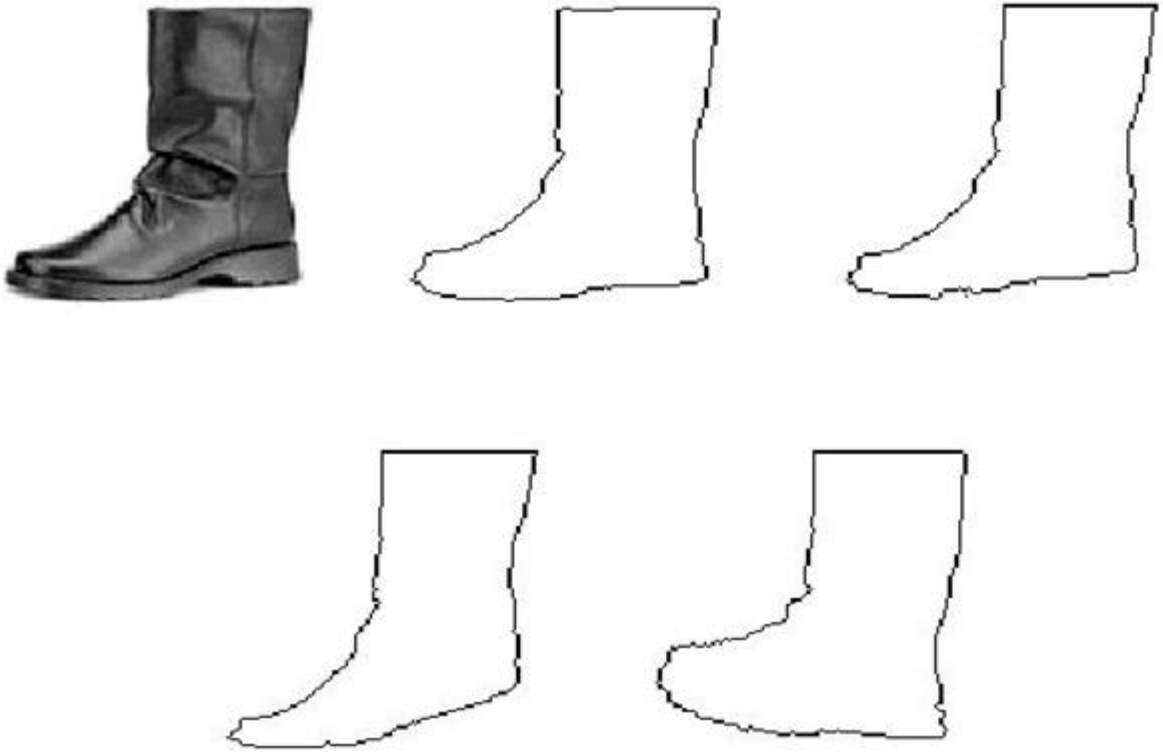


CSE 473
Pattern Recognition

Template Matching

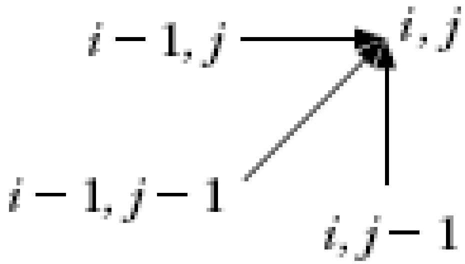


The Edit Distance

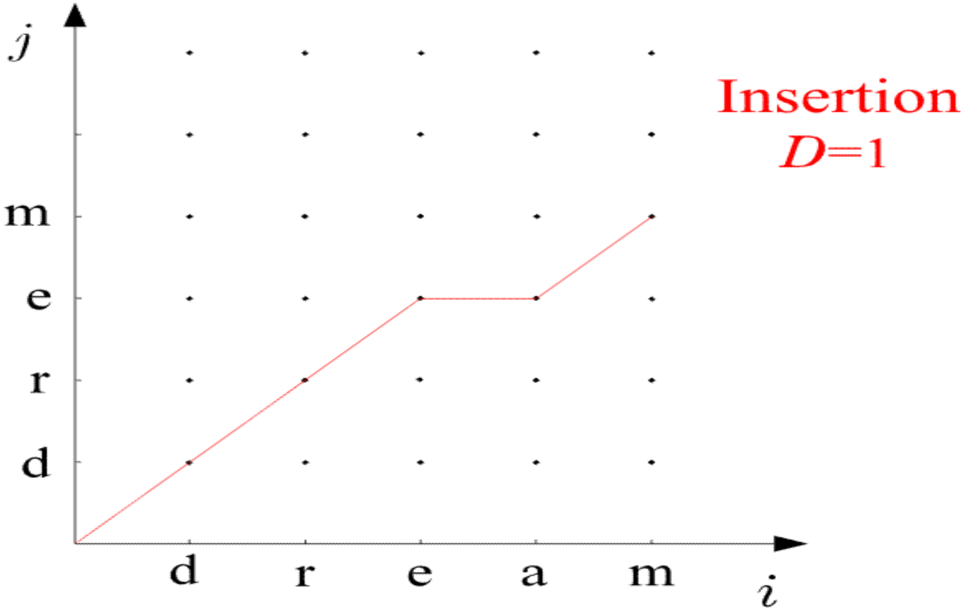
- Cost *D*(0,0) = 0,
- Complete path is searched
- Allowable predecessors and costs
  - $(i-1, j-1) \rightarrow (i, j)$

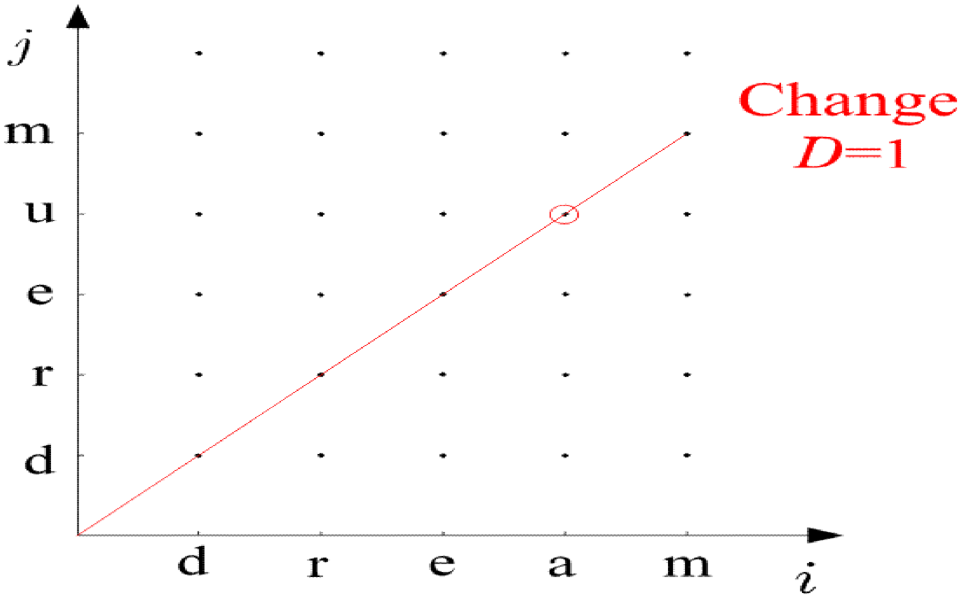
    $$d\left(i, j \mid i-1, j-1\right) = \begin{cases} 0, & \text{if } t(i) = r(j) \\ 1, & t(i) \neq r(j) \end{cases}$$

  - Horizontal $d(i, j \mid i-1, j) = 1$
  - Vertical $d(i, j \mid i, j-1) = 1$

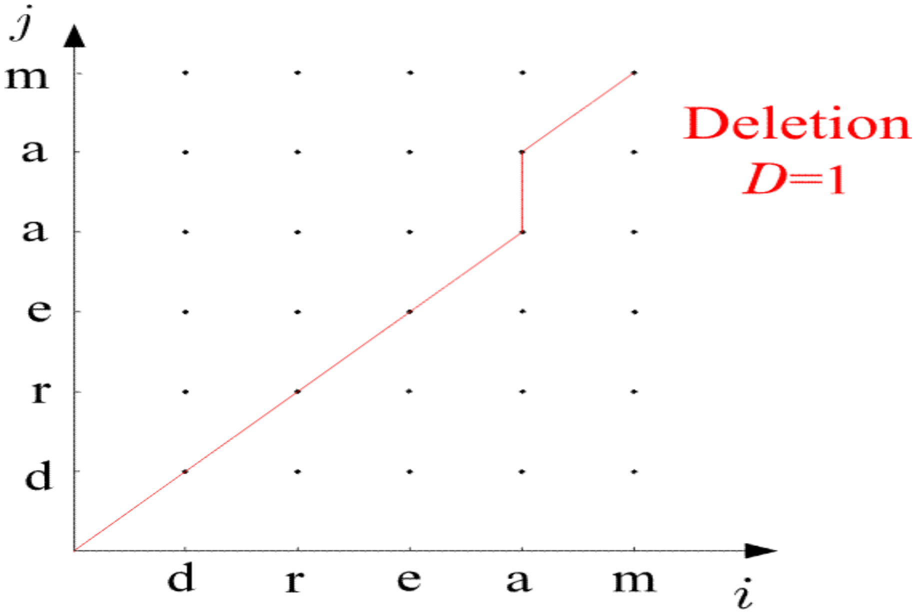


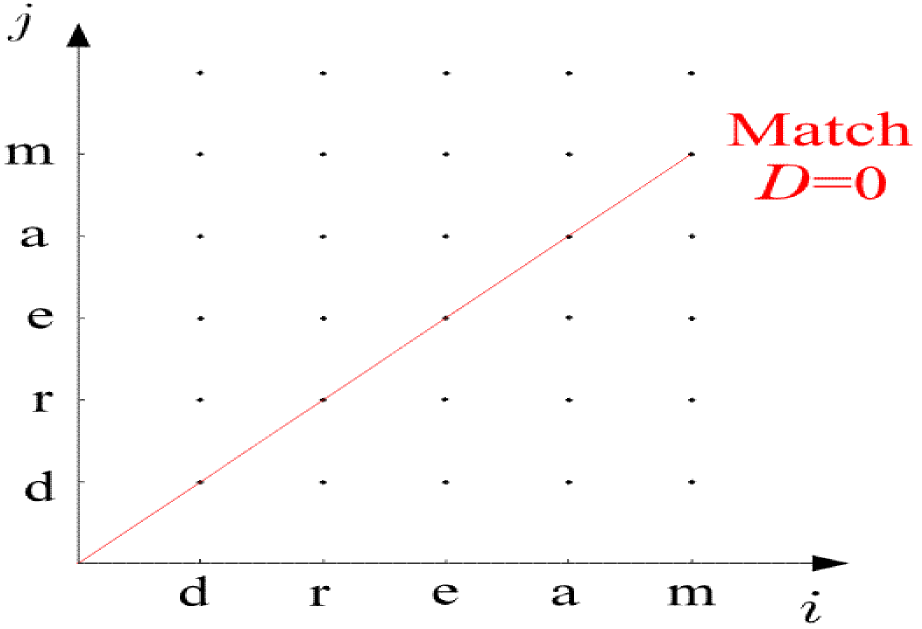
- Examples:





- Examples:

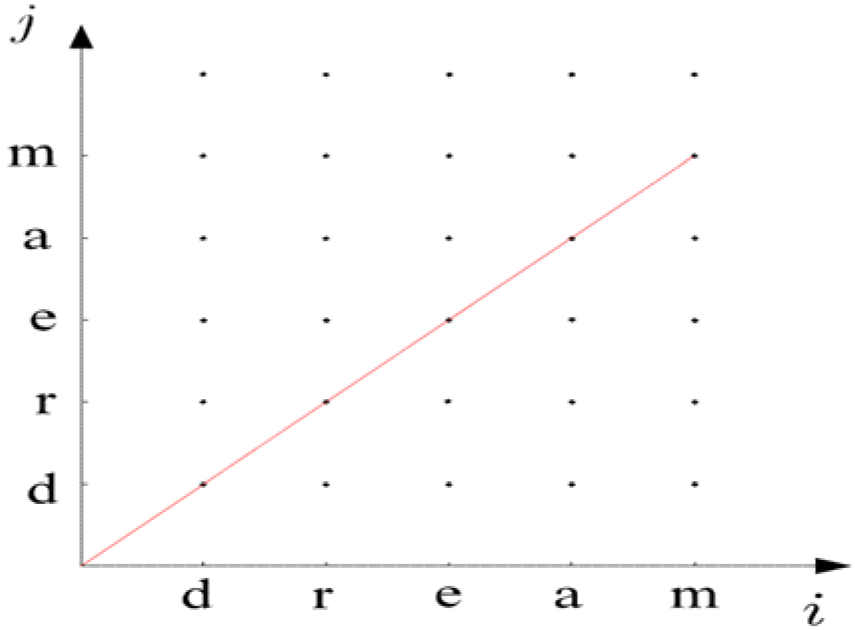




The Edit Distance

- 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}
$D(A,B)=D(I,J)$
```



Application of TM in Speech Recognition

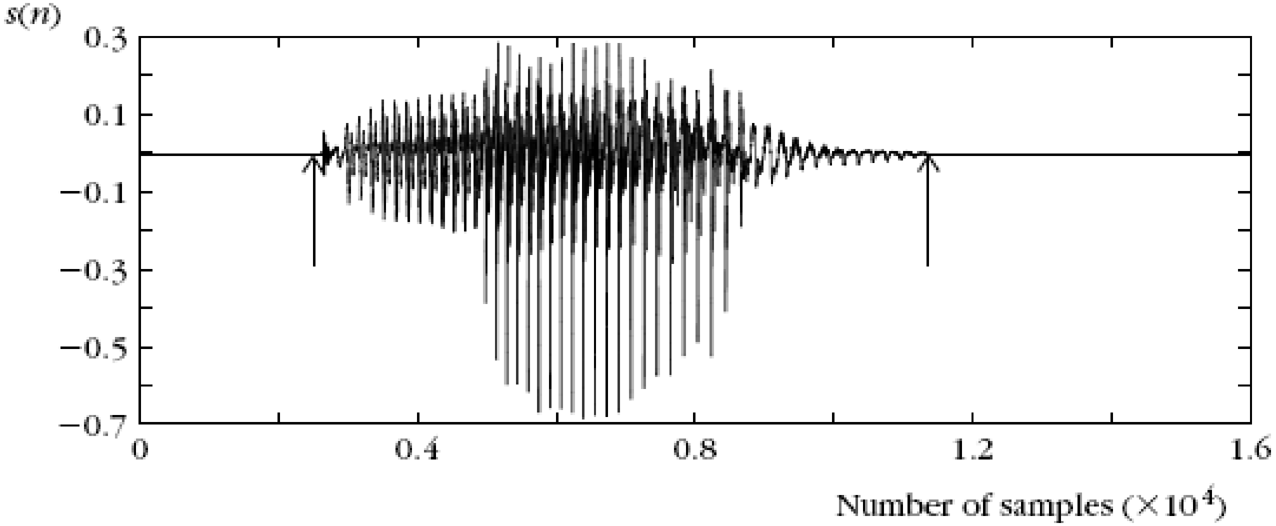
- A number of variations
  - Speaker Independent Speech Recognition
  - Speaker Dependent Speech Recognition
  - Continuous Speech Recognition
  - Isolated word recognition (IWR)

Application of TM in IWR

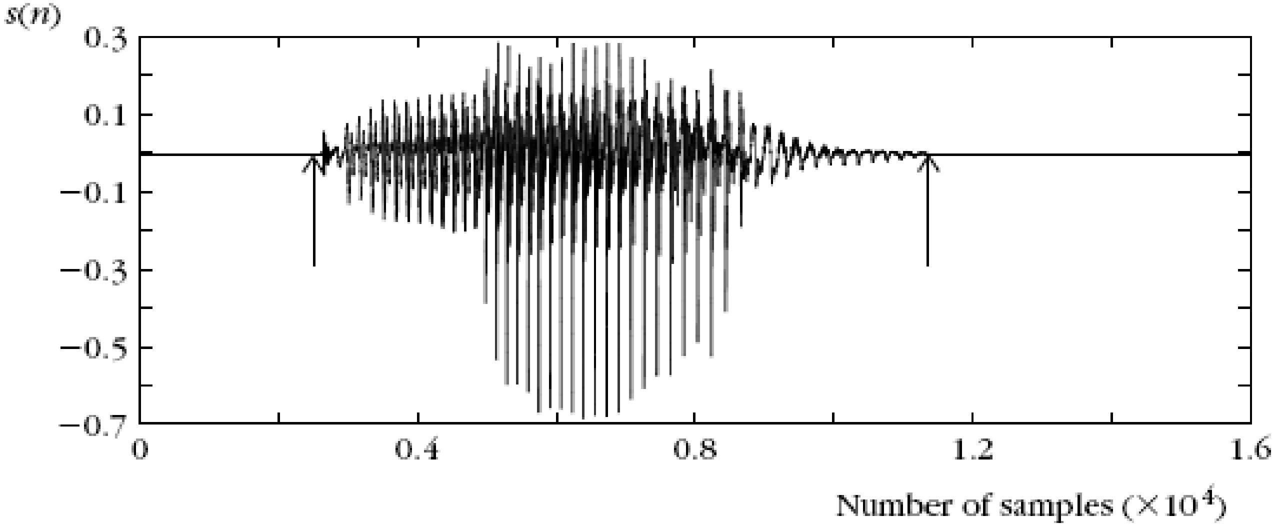
- The goal:
    - Given a number of known spoken words in a data base (reference patterns)
    - find the best match of an unknown spoken word (test pattern).

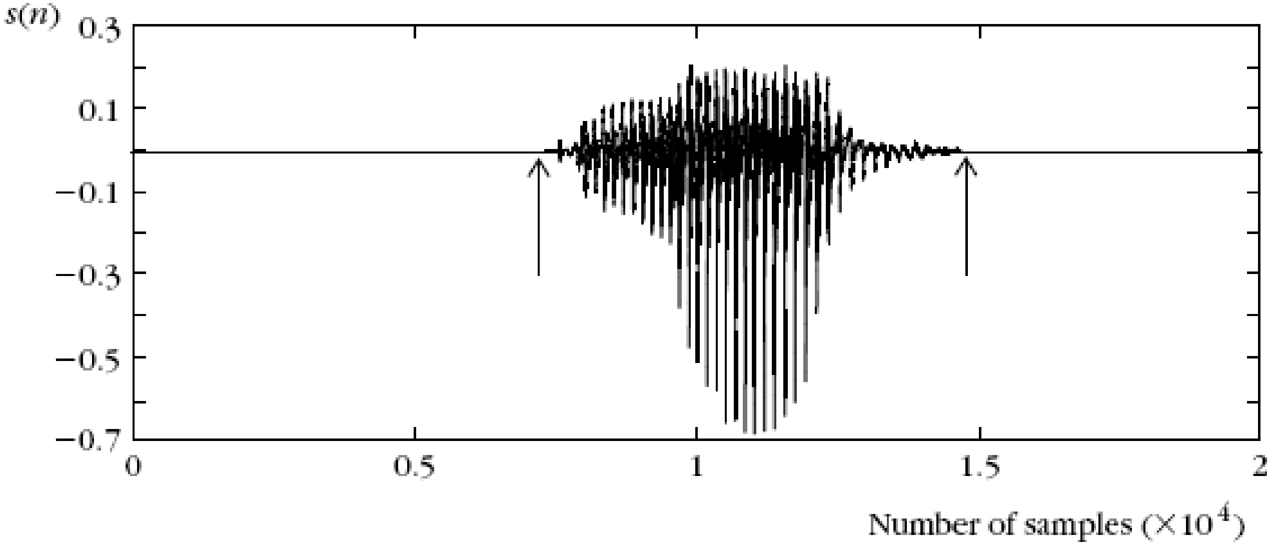
- Procedure:
    - compare the test word against reference words

Application of TM in IWR



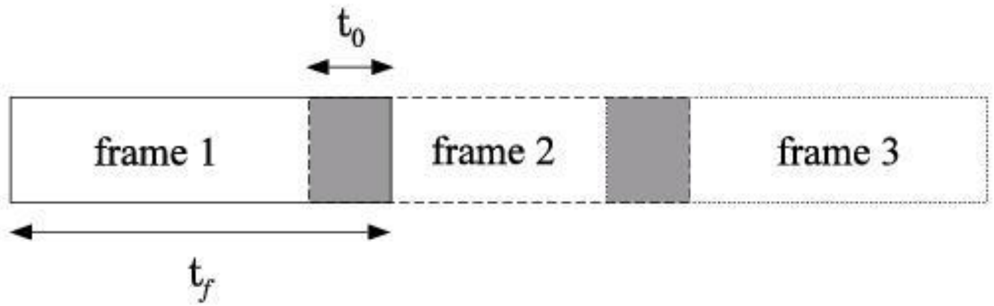
Application of TM in IWR





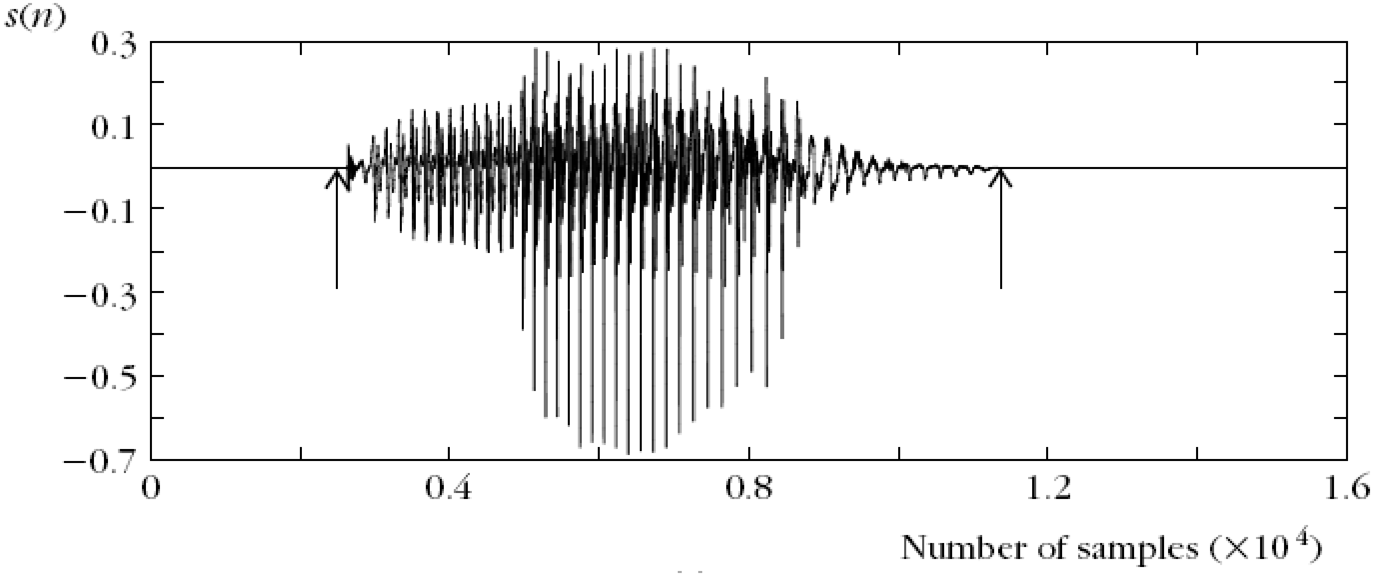
Application of TM in IWR

- 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.



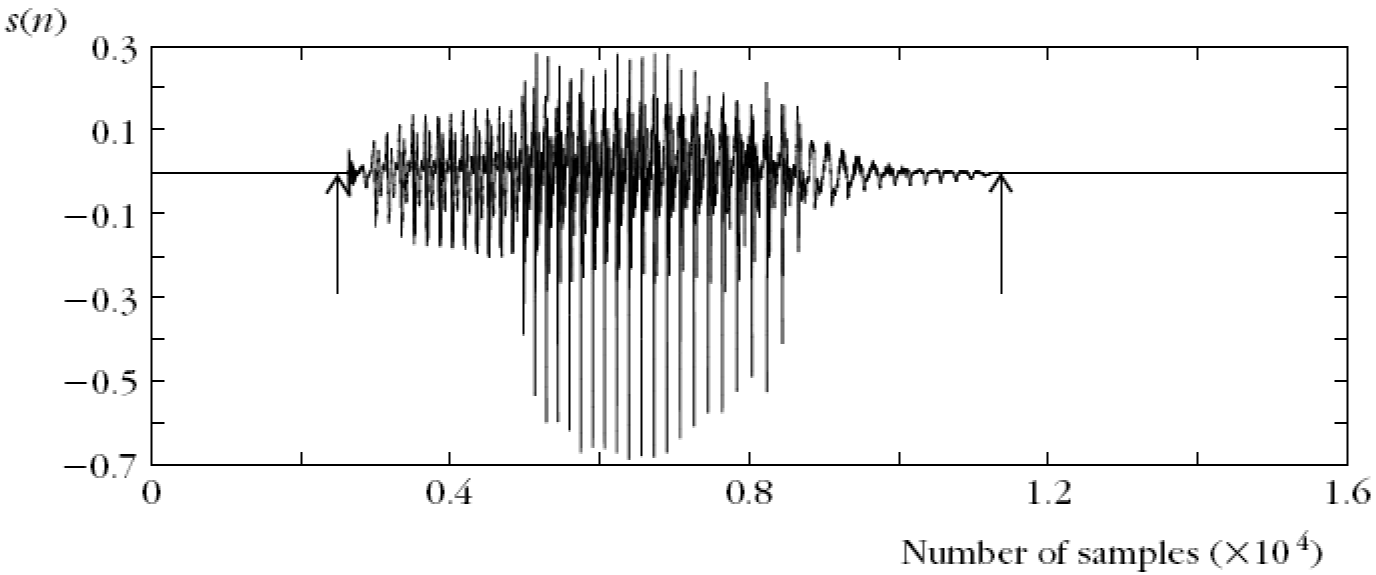
Application of TM in IWR

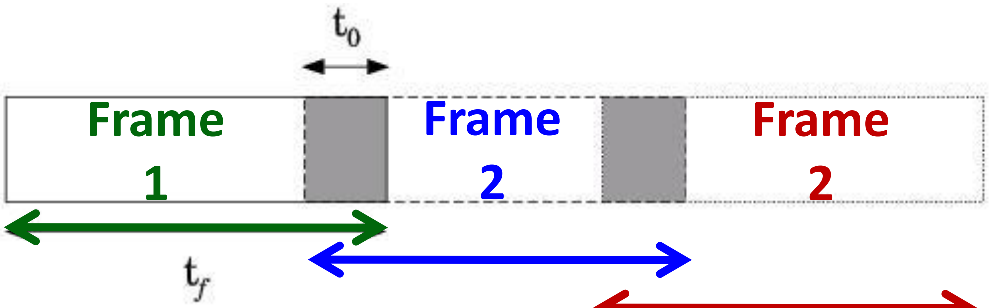
- The procedure:
  - Sample a speech segment from a microphone:



Application of TM in IWR

- The procedure:





t_0

Frame 1

Frame 2

Frame 2

t_f

$t_f = 512$

$t_0 = 100$

- each frame is represented by a vector of 512 samples

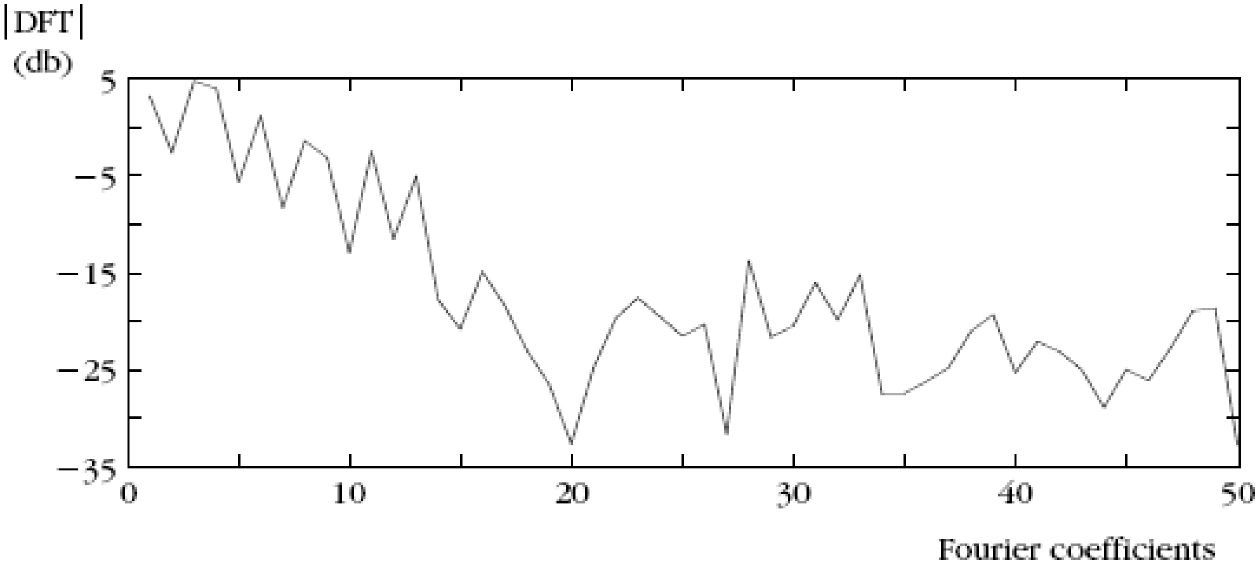
$$\underline{r}(i) = \begin{bmatrix} x_i(0) \\ x_i(1) \\ \ldots \\ \ldots \\ x_i(511) \end{bmatrix}, \; i = 1, \ldots, I \qquad \underline{t}(j) = \begin{bmatrix} x_j(0) \\ x_j(1) \\ \ldots \\ \ldots \\ x_j(511) \end{bmatrix}, \; j = 1, \ldots, J$$

- convert them to DFT

$$DFT(\underline{r}(i)) = DFT(\begin{bmatrix} x_i(0) \\ x_i(1) \\ \ldots \\ \ldots \\ x_i(511) \end{bmatrix}) = \begin{bmatrix} X_i(0) \\ X_i(1) \\ \ldots \\ \ldots \\ X_i(511) \end{bmatrix}$$

$$DFT(\underline{t}(j)) = DFT(\begin{bmatrix} x_i(0) \\ x_i(1) \\ \ldots \\ \ldots \\ x_i(511) \end{bmatrix}) = \begin{bmatrix} X_i(0) \\ X_i(1) \\ \ldots \\ \ldots \\ X_i(511) \end{bmatrix}$$

- convert them to DFT



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

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

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

$$\underline{r}(i)=\begin{bmatrix} X_i(0) \\ X_i(1) \\ \ldots \\ \ldots \\ X_i(\ell-1) \end{bmatrix},\ i=1,\ \ldots,\ I \qquad \underline{t}(j)=\begin{bmatrix} X_j(0) \\ X_j(1) \\ \ldots \\ \ldots \\ X_j(\ell-1) \end{bmatrix},\ j=1,\ \ldots,\ J$$

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

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

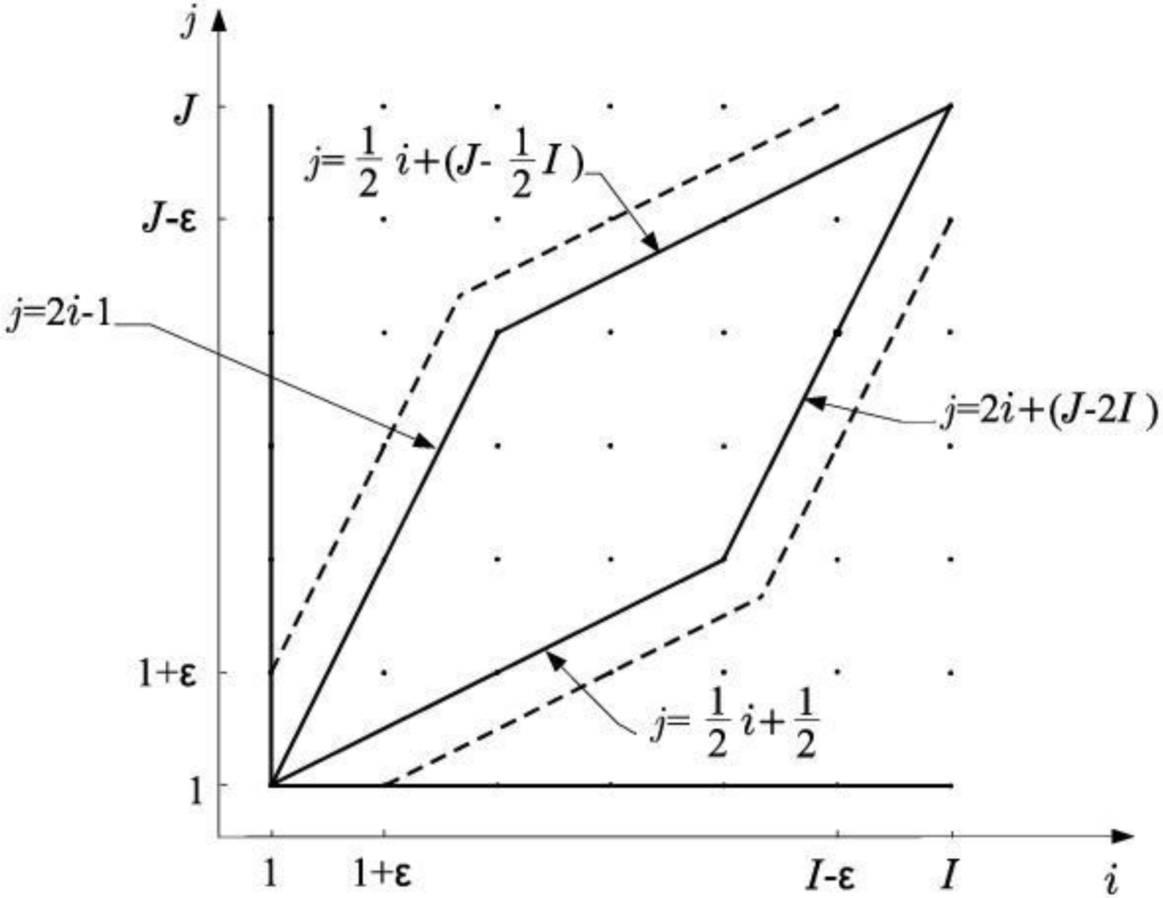
- Choose a cost function associated with each node across a path, e.g., the Euclidean distance

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

- find the optimal path in the grid
- Match the test pattern to the reference pattern associated with the optimal path

- Prior to performing the math one has to choose:
    - end point constraints
    - global constraints
    - local constraints
    - distance

- 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.



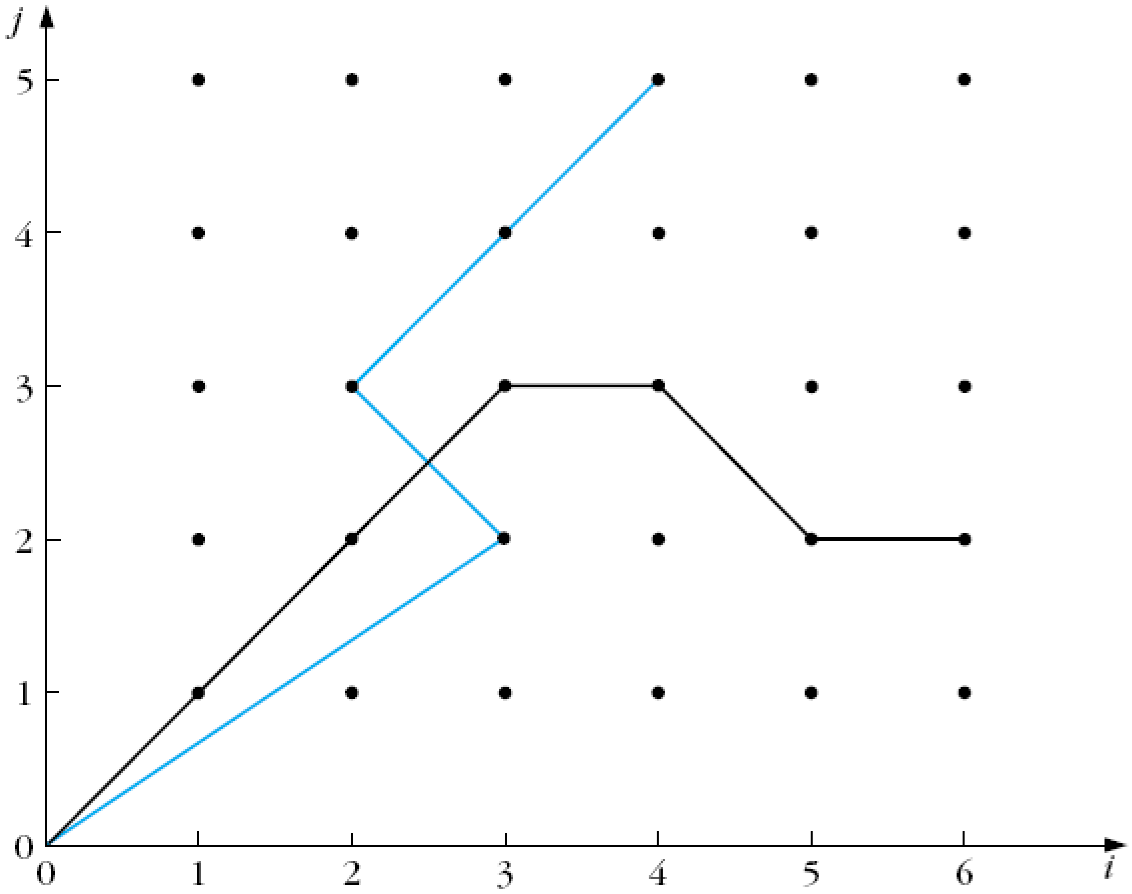
- The local constraints: monotonic path

$$i_{k-1} \leq i_k \quad \text{and} \quad j_{k-1} \leq j_k$$

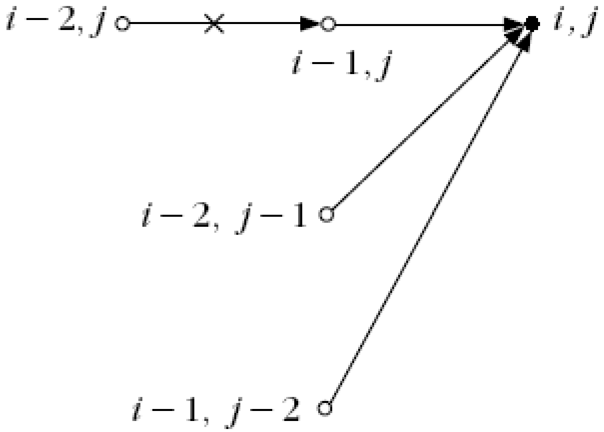
- The local constraints: monotonic path

$$i_{k-1} \leq i_k \quad \text{and} \quad j_{k-1} \leq j_k$$

- Non-monotonic path

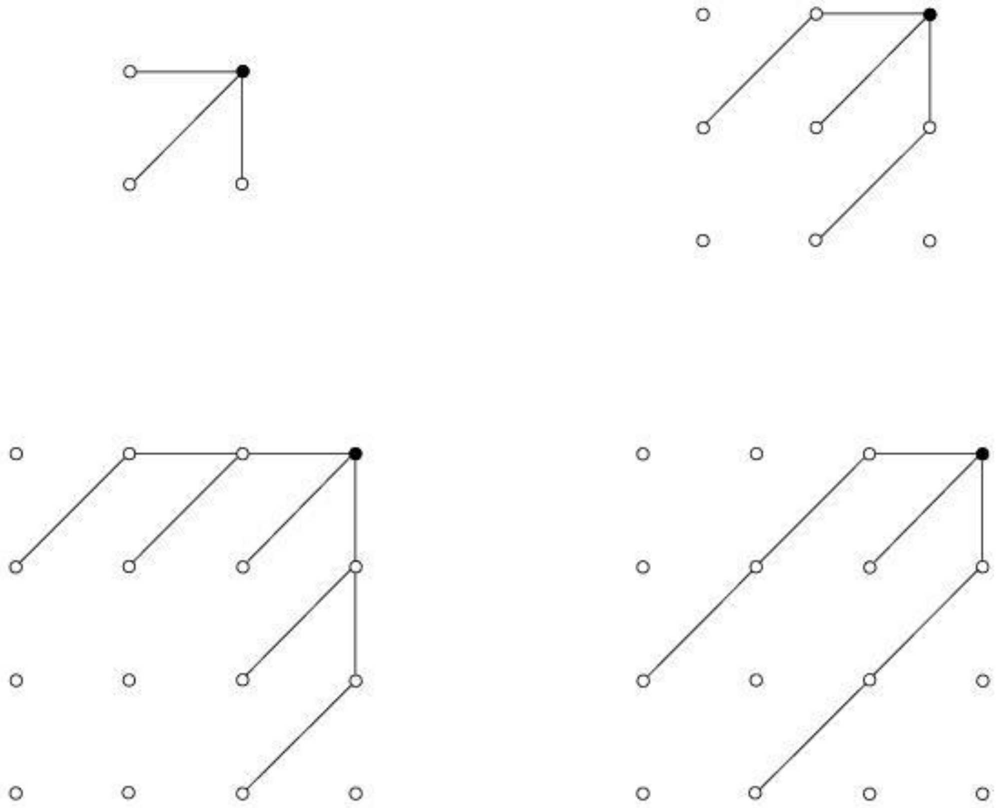


- The local constraints: Defining the type of transitions allowed between the nodes of the grid.



Itakura local constraints

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



Sakoe and Chiba local constraints

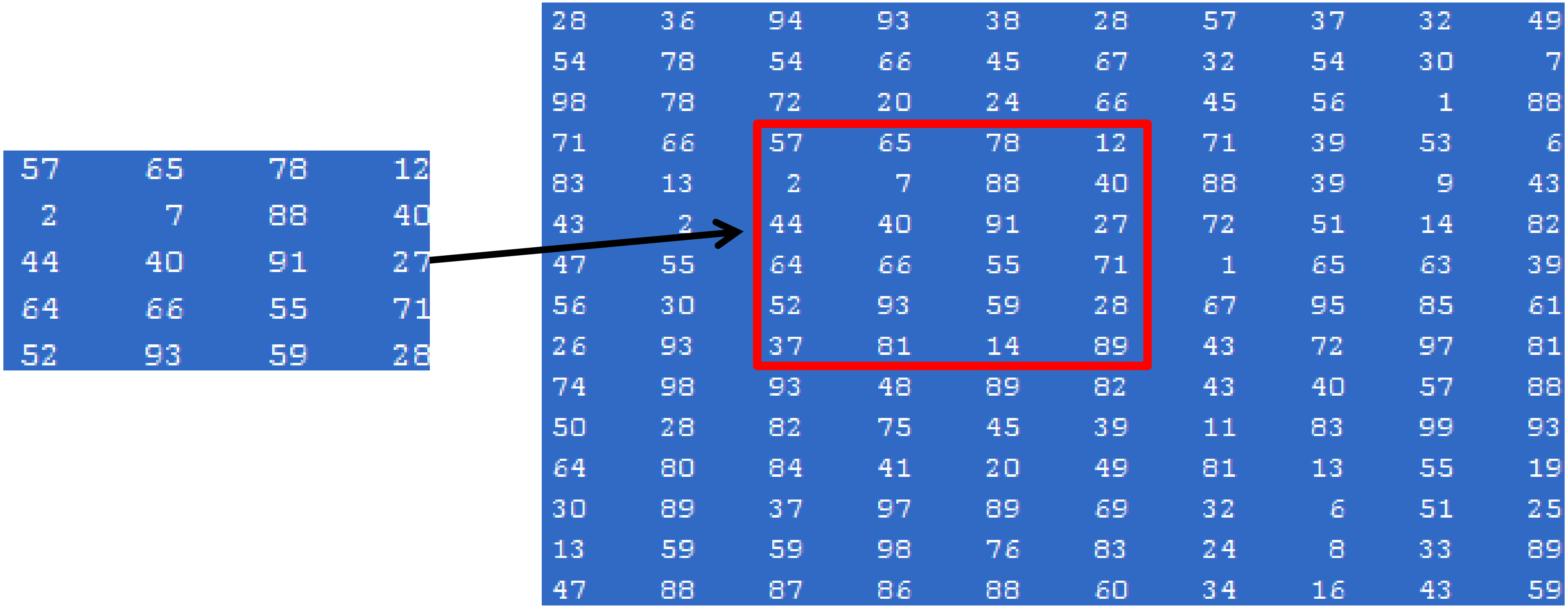
- cost function:
  - Euclidean distance
  - only node distance

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

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

Correlation based TM

- Goal: to find whether a specific known reference pattern resides within a given block of data.

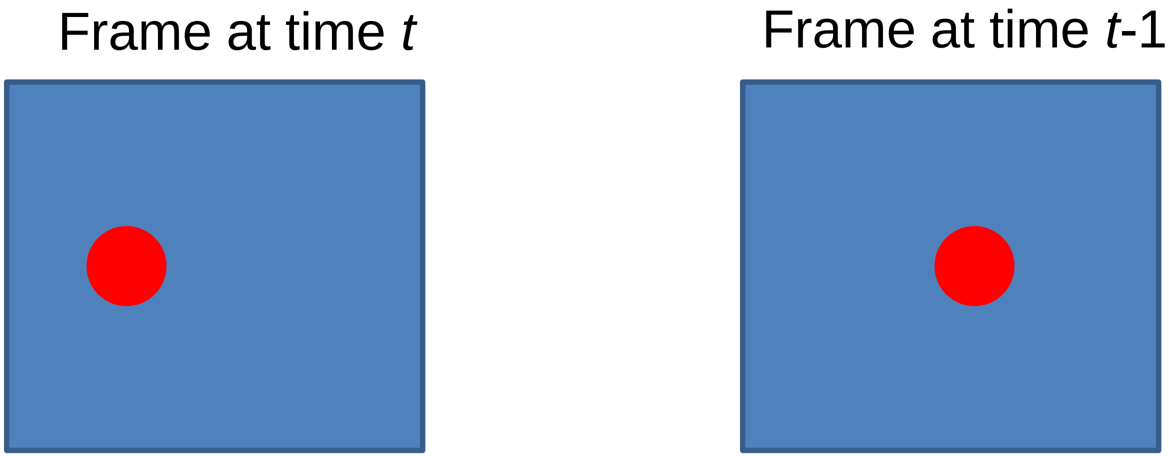


57	65	78	12
2	7	88	40
44	40	91	27
64	66	55	71
52	93	59	28

28	36	94	93	38	28	57	37	32	49
54	78	54	66	45	67	32	54	30	7
98	78	72	20	24	66	45	56	1	88
71	66	57	65	78	12	71	39	53	6
83	13	2	7	88	40	88	39	9	43
43	2	44	40	91	27	72	51	14	82
47	55	64	66	55	71	1	65	63	39
56	30	52	93	59	28	67	95	85	61
26	93	37	81	14	89	43	72	97	81
74	98	93	48	89	82	43	40	57	88
50	28	82	75	45	39	11	83	99	93
64	80	84	41	20	49	81	13	55	19
30	89	37	97	89	69	32	6	51	25
13	59	59	98	76	83	24	8	33	89
47	88	87	86	88	60	34	16	43	59

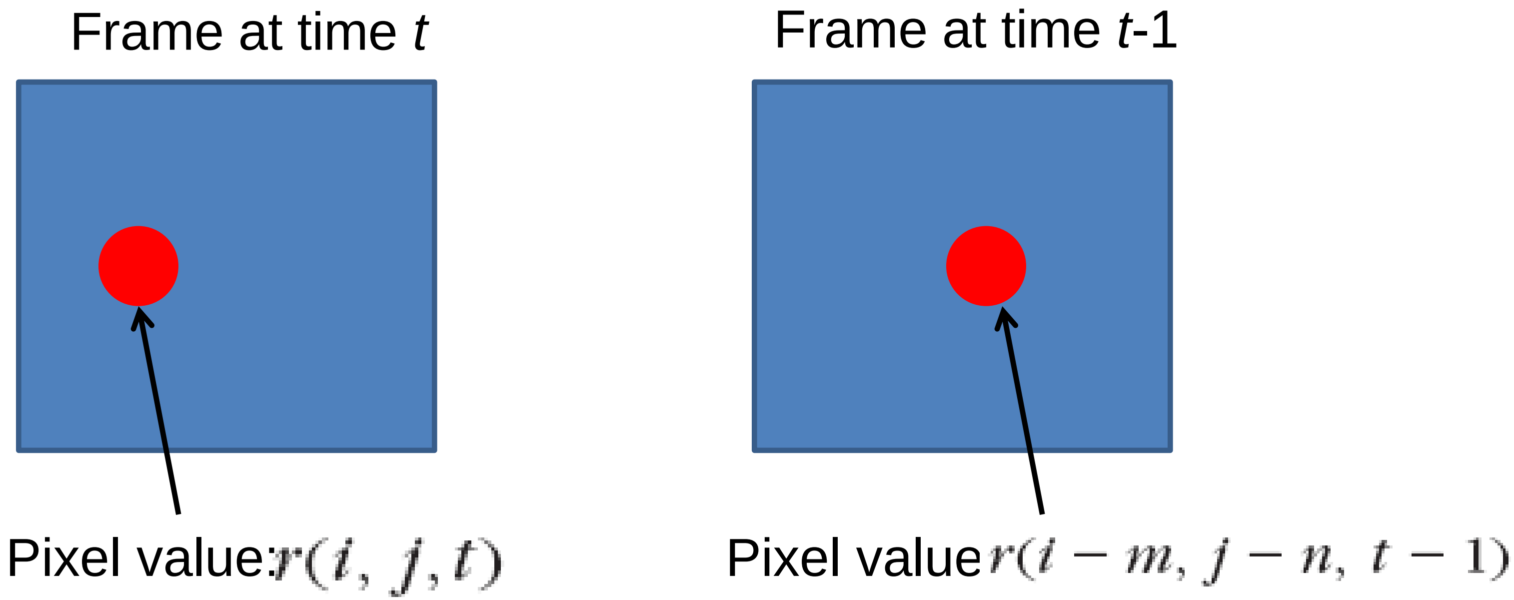
Correlation based TM

- Application: target detection, robot vision, video coding.



Correlation based TM

- Application: target detection, robot vision, video coding.



Frame at time *t*

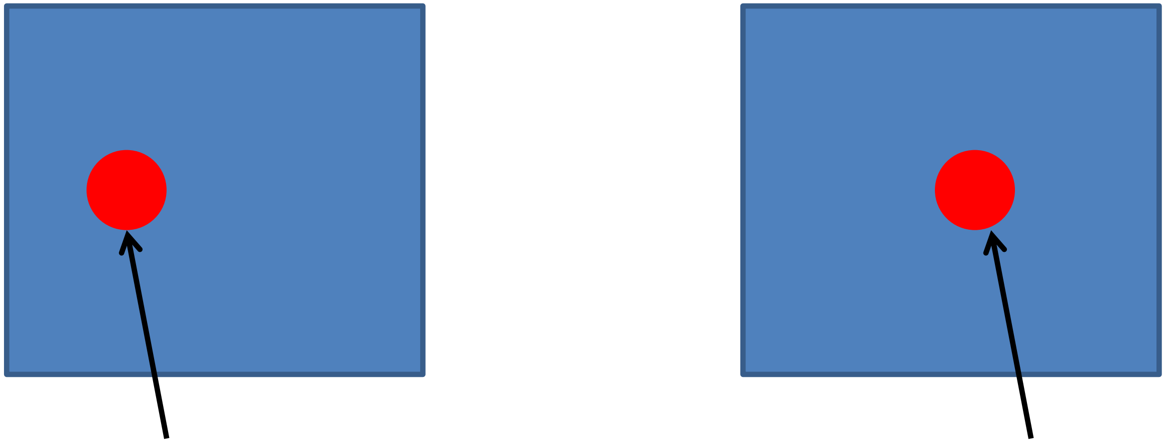
Frame at time *t*-1

Pixel value: $r(i, j, t)$

Pixel value $r(i-m, j-n, t-1)$

Correlation based TM

- Application: target detection, robot vision, video coding.



Frame at time *t*

Frame at time *t*-1

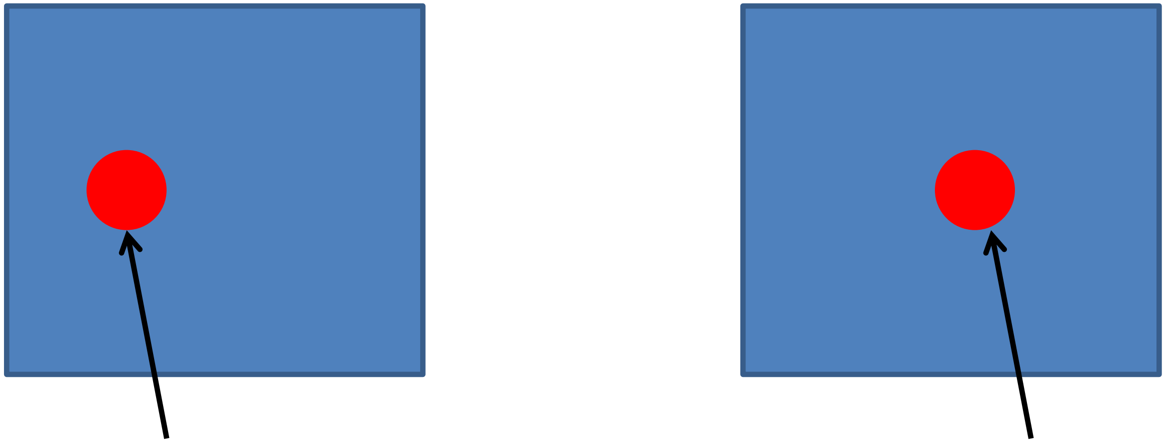
Pixel value: $r(i, j, t)$

Pixel value $r(i-m, j-n, t-1)$

Difference $e(i, j, t) = r(i, j, t) - r(i-m, j-n, t-1)$

Correlation based TM

- Application: target detection, robot vision, video coding.



Frame at time *t*

Frame at time *t*-1

Pixel value: $r(i, j, t)$

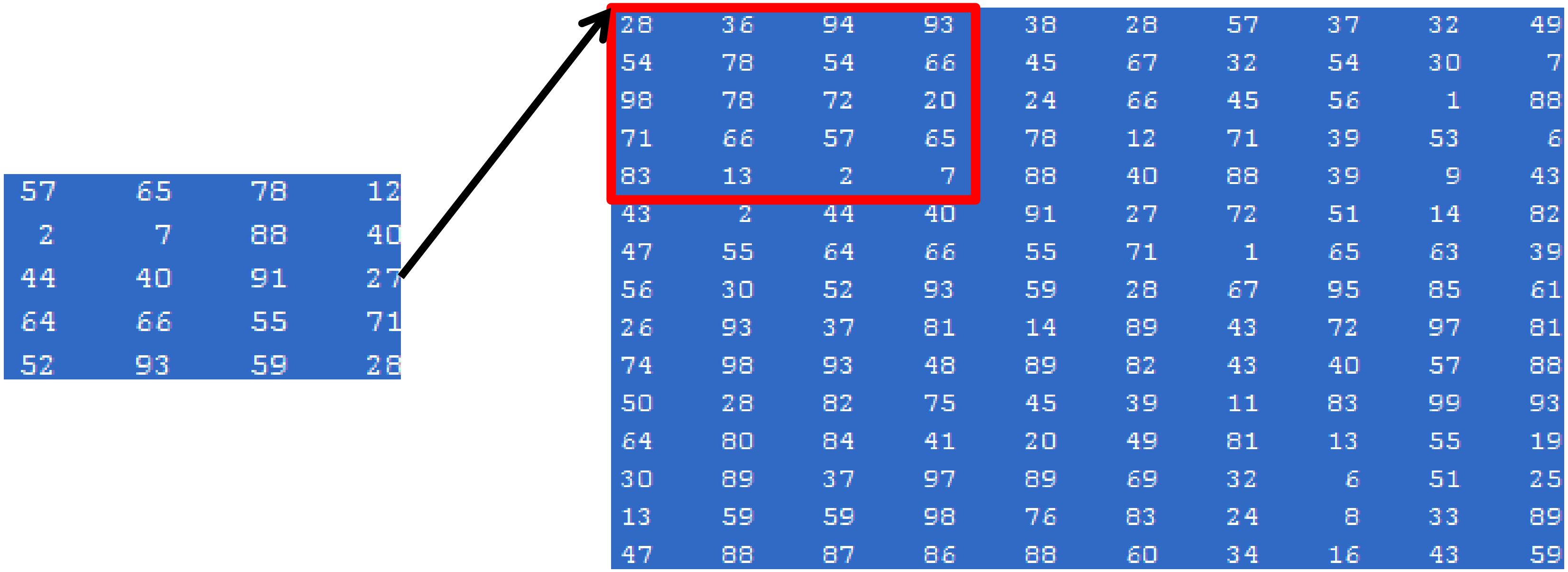
Pixel value $r(i - m, j - n, t - 1)$

Difference $e(i, j, t) = r(i, j, t) - r(i - m, j - n, t - 1)$

- We need to encode only the difference

Correlation based TM

- 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.

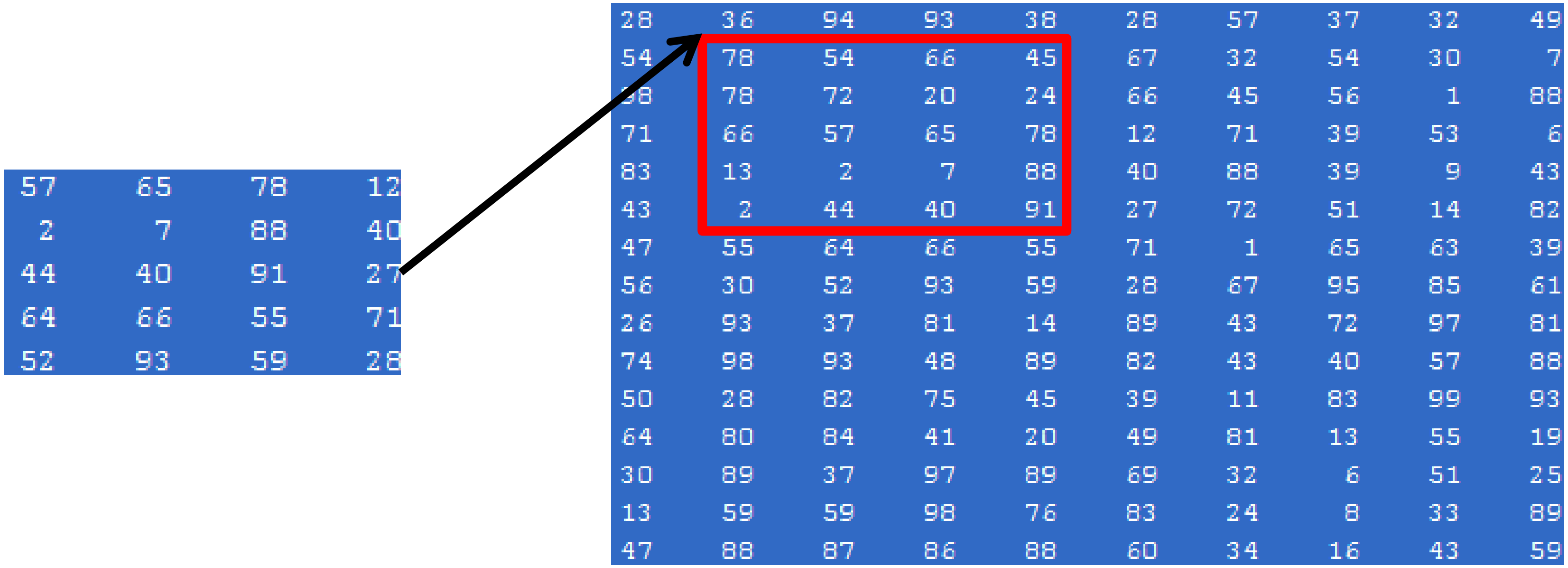


57	65	78	12
2	7	88	40
44	40	91	27
64	66	55	71
52	93	59	28

28	36	94	93	38	28	57	37	32	49
54	78	54	66	45	67	32	54	30	7
98	78	72	20	24	66	45	56	1	88
71	66	57	65	78	12	71	39	53	6
83	13	2	7	88	40	88	39	9	43
43	2	44	40	91	27	72	51	14	82
47	55	64	66	55	71	1	65	63	39
56	30	52	93	59	28	67	95	85	61
26	93	37	81	14	89	43	72	97	81
74	98	93	48	89	82	43	40	57	88
50	28	82	75	45	39	11	83	99	93
64	80	84	41	20	49	81	13	55	19
30	89	37	97	89	69	32	6	51	25
13	59	59	98	76	83	24	8	33	89
47	88	87	86	88	60	34	16	43	59

Correlation based TM

- 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.

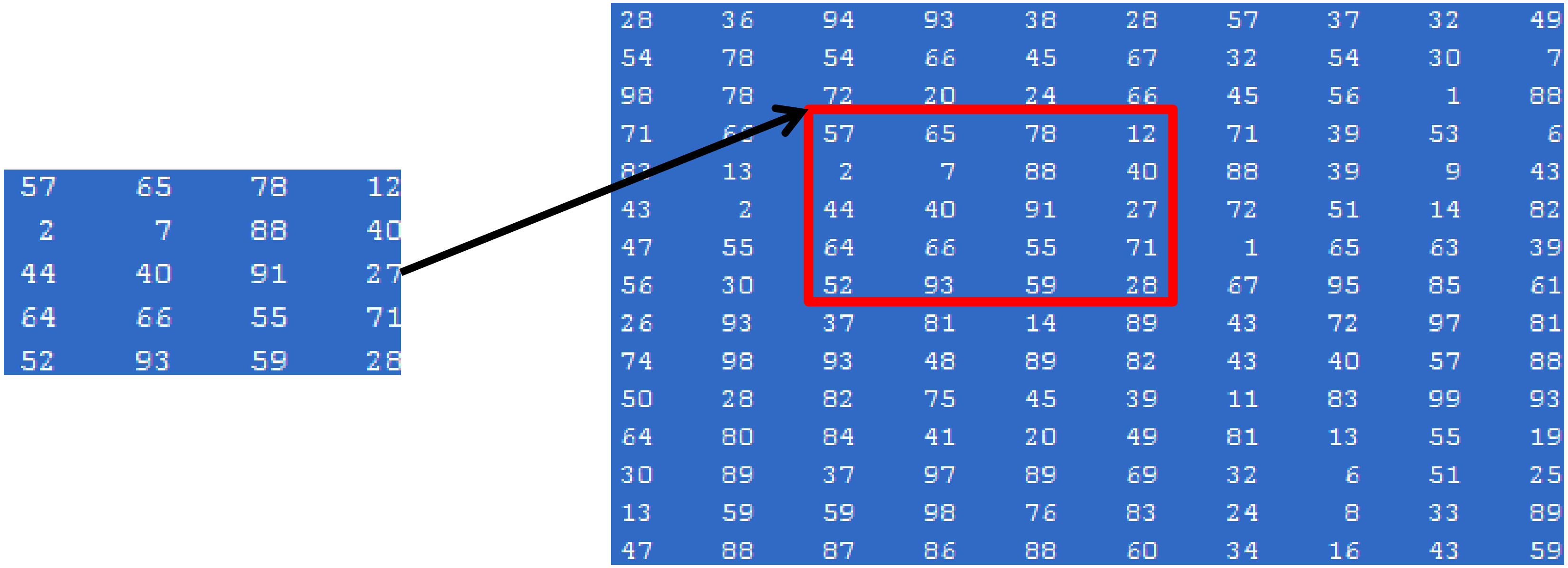


57	65	78	12
2	7	88	40
44	40	91	27
64	66	55	71
52	93	59	28

28	36	94	93	38	28	57	37	32	49
54	78	54	66	45	67	32	54	30	7
98	78	72	20	24	66	45	56	1	88
71	66	57	65	78	12	71	39	53	6
83	13	2	7	88	40	88	39	9	43
43	2	44	40	91	27	72	51	14	82
47	55	64	66	55	71	1	65	63	39
56	30	52	93	59	28	67	95	85	61
26	93	37	81	14	89	43	72	97	81
74	98	93	48	89	82	43	40	57	88
50	28	82	75	45	39	11	83	99	93
64	80	84	41	20	49	81	13	55	19
30	89	37	97	89	69	32	6	51	25
13	59	59	98	76	83	24	8	33	89
47	88	87	86	88	60	34	16	43	59

Correlation based TM

- 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.

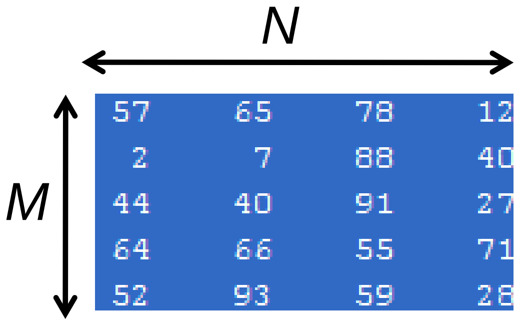


57	65	78	12
2	7	88	40
44	40	91	27
64	66	55	71
52	93	59	28

28	36	94	93	38	28	57	37	32	49
54	78	54	66	45	67	32	54	30	7
98	78	72	20	24	66	45	56	1	88
71	66	57	65	78	12	71	39	53	6
82	13	2	7	88	40	88	39	9	43
43	2	44	40	91	27	72	51	14	82
47	55	64	66	55	71	1	65	63	39
56	30	52	93	59	28	67	95	85	61
26	93	37	81	14	89	43	72	97	81
74	98	93	48	89	82	43	40	57	88
50	28	82	75	45	39	11	83	99	93
64	80	84	41	20	49	81	13	55	19
30	89	37	97	89	69	32	6	51	25
13	59	59	98	76	83	24	8	33	89
47	88	87	86	88	60	34	16	43	59

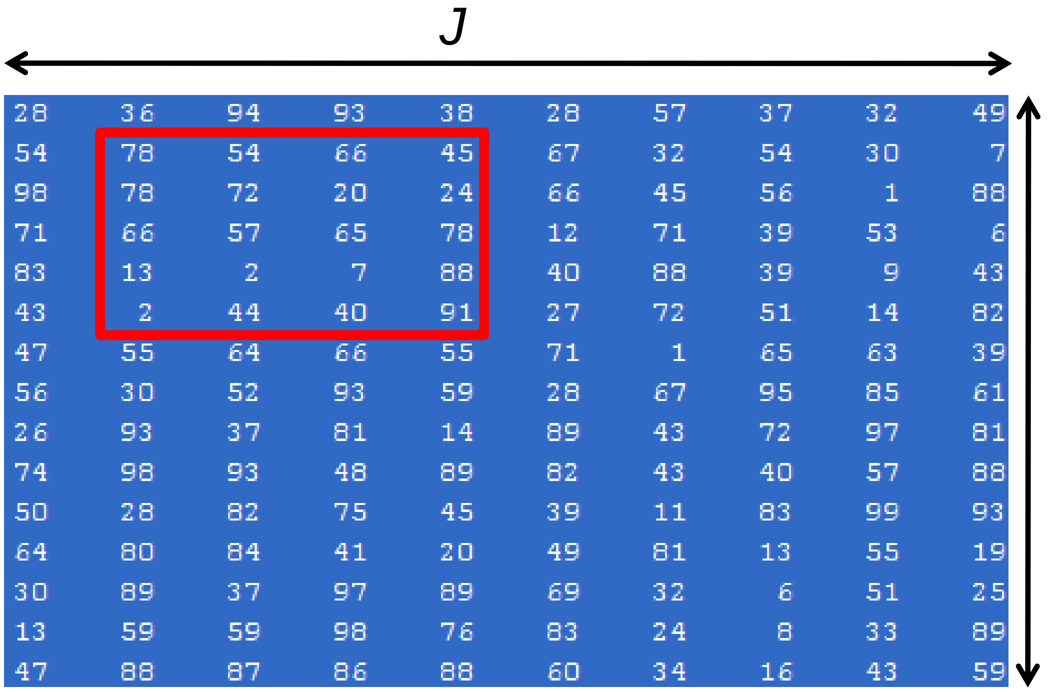
Correlation In Image Matching

- 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 $t(i,j)$.



57	65	78	12
2	7	88	40
44	40	91	27
64	66	55	71
52	93	59	28

Reference, *r*



28	36	94	93	38	28	57	37	32	49
54	78	54	66	45	67	32	54	30	7
98	78	72	20	24	66	45	56	1	88
71	66	57	65	78	12	71	39	53	6
83	13	2	7	88	40	88	39	9	43
43	2	44	40	91	27	72	51	14	82
47	55	64	66	55	71	1	65	63	39
56	30	52	93	59	28	67	95	85	61
26	93	37	81	14	89	43	72	97	81
74	98	93	48	89	82	43	40	57	88
50	28	82	75	45	39	11	83	99	93
64	80	84	41	20	49	81	13	55	19
30	89	37	97	89	69	32	6	51	25
13	59	59	98	76	83	24	8	33	89
47	88	87	86	88	60	34	16	43	59

Test, *t*

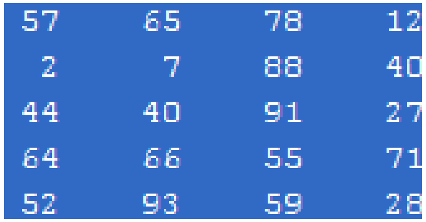
Correlation In Image Matching

- Compute the distance:

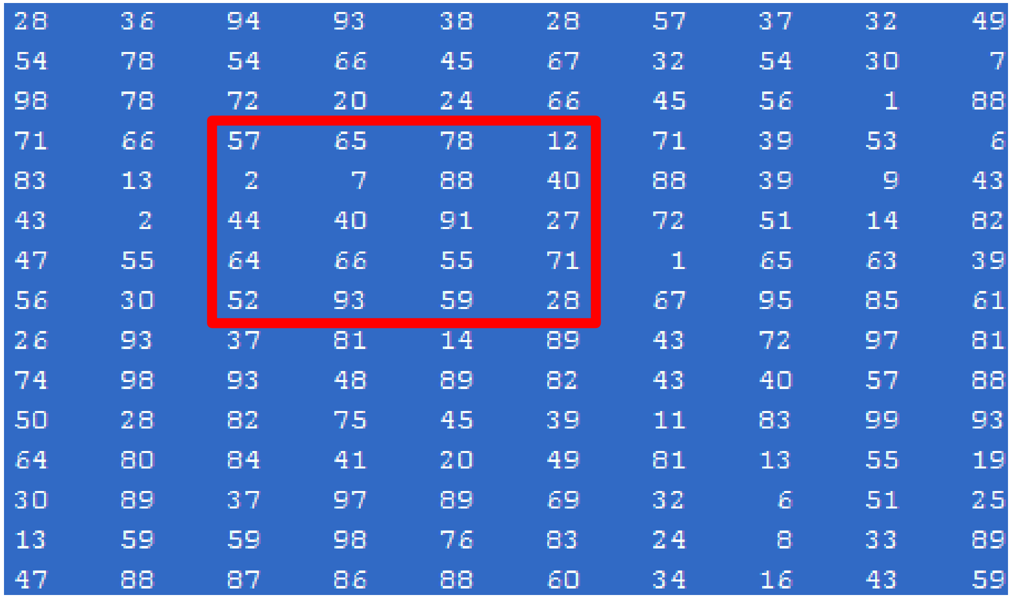
$$D(m,n) = \sum_{i=m}^{m+M-1} \sum_{j=n}^{n+N-1} |t(i,j) - r(i-m, j-n)|^2$$

for every (m,n).

- For all (m,n), compute the minimum.



57	65	78	12
2	7	88	40
44	40	91	27
64	66	55	71
52	93	59	28



28	36	94	93	38	28	57	37	32	49
54	78	54	66	45	67	32	54	30	7
98	78	72	20	24	66	45	56	1	88
71	66	57	65	78	12	71	39	53	6
83	13	2	7	88	40	88	39	9	43
43	2	44	40	91	27	72	51	14	82
47	55	64	66	55	71	1	65	63	39
56	30	52	93	59	28	67	95	85	61
26	93	37	81	14	89	43	72	97	81
74	98	93	48	89	82	43	40	57	88
50	28	82	75	45	39	11	83	99	93
64	80	84	41	20	49	81	13	55	19
30	89	37	97	89	69	32	6	51	25
13	59	59	98	76	83	24	8	33	89
47	88	87	86	88	60	34	16	43	59

Correlation In Image Matching

- The equation

$$D(m,n) = \sum_{i=m}^{m+M-1} \sum_{j=n}^{n+N-1} |t(i,j) - r(i-m, j-n)|^2$$

can be written as

$$D(m,n) = \sum_{i} \sum_{j} |t(i,j)|^2 + \sum_{i} \sum_{j} |r(i,j)|^2$$

$$- 2 \sum_{i} \sum_{j} t(i,j) r(i-m, j-n)$$

Correlation In Image Matching

- In the equation

$$D(m,n) = \sum_i \sum_j |t(i,j)|^2 + \sum_i \sum_j |r(i,j)|^2$$

$$- 2 \sum_i \sum_j t(i,j) r(i-m, j-n)$$

shaded terms are constant

provided pixel levels do not change much across the test image

Correlation In Image Matching

$$D(m,n) = \boxed{\sum_i \sum_j |t(i,j)|^2} + \boxed{\sum_i \sum_j |r(i,j)|^2}$$

$$- 2 \sum_i \sum_j t(i,j) r(i-m, j-n)$$

- Canceling out the shaded terms, find point (*m*, *n*) that maximize:

$$c(m,n) = \sum_i \sum_j t(i,j) r(i-m, j-n)$$

Correlation In Image Matching

$$c(m,n) = \sum_{i} \sum_{j} t(i,j) r(i-m, j-n)$$

- *c* (*m*, *n*) is no longer a difference term
- This is called cross correlation

Correlation In Image Matching

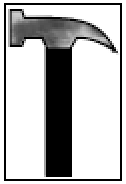
$$c(m,n) = \sum_{i}\sum_{j} t(i,j)r(i-m,j-n)$$

- In case gray level variation is valid, normalize:

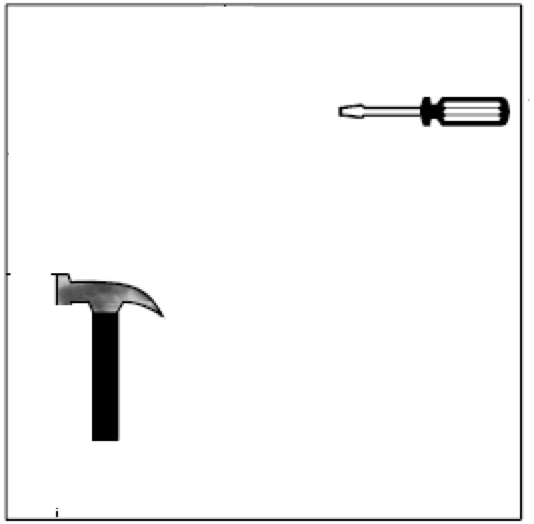
$$c_N(m,n) = \frac{c(m,n)}{\sqrt{\sum_i \sum_j |t(i,j)|^2 \sum_i \sum_j |r(i,j)|^2}}$$

Correlation In Image Matching

- Example:



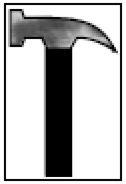
Reference, *r*



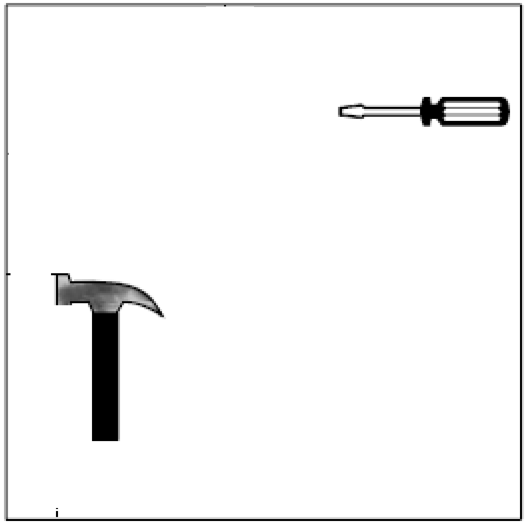
Test Image, *t*

Correlation In Image Matching

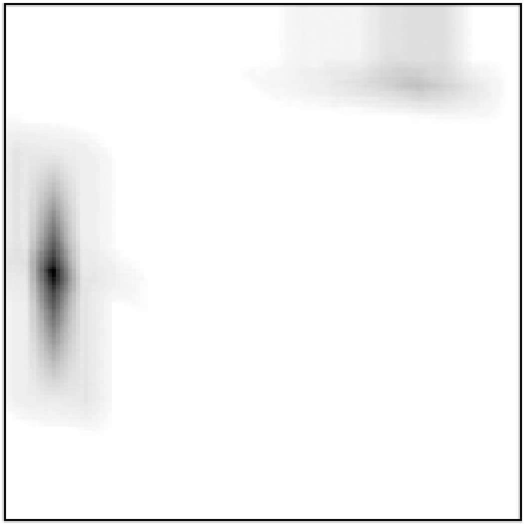
- Example:



Reference, *r*



Test Image, *t*



Correlation Image

Computation Considerations in Correlation Based TM (1)

- Find *c* (*m*,*n*) at every pixel

$$c(m,n) = \sum_i \sum_j t(i,j) r(i-m, j-n)$$

- This equation looks like convolution operation
- Alternate is to calculate in the frequency domain

Computation Considerations in Correlation Based TM (1)

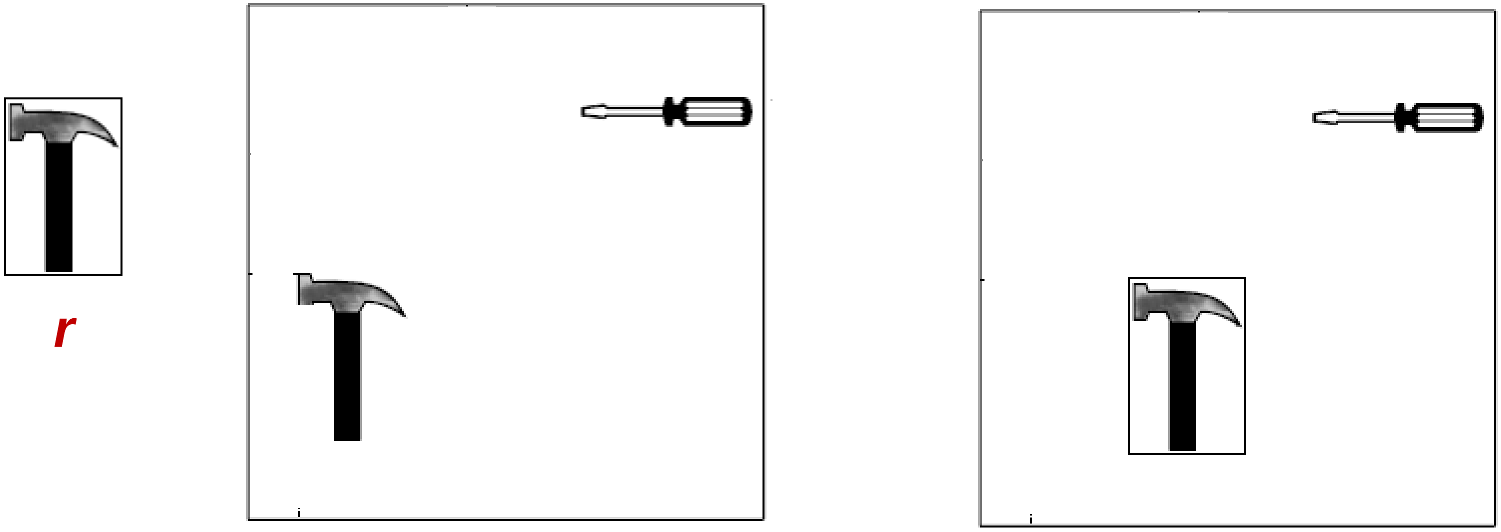
- The frequency domain representation of

$$c(m,n) = \sum_i \sum_j t(i,j)r(i-m,j-n)$$

is $c = IDFT(DFT(t)DFT(r))$

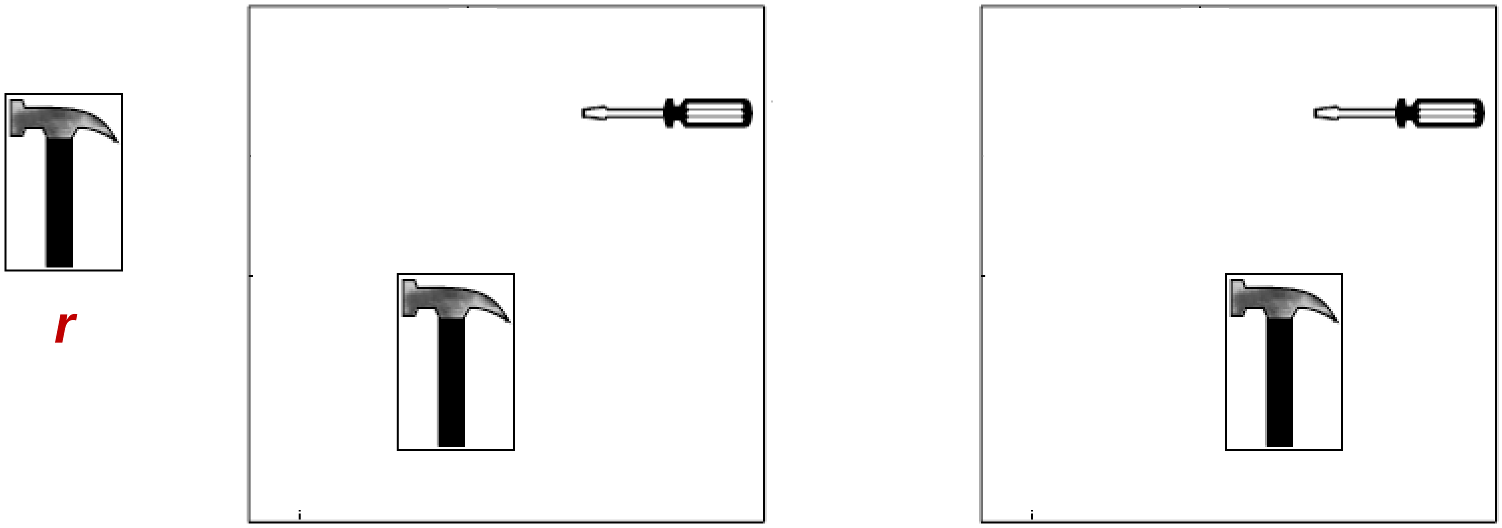
Computation Considerations in Correlation Based TM (2)

- Limit the search space
  - Search only in the area of [$-p$, p] X [$-p$, p] centered at (x, y)



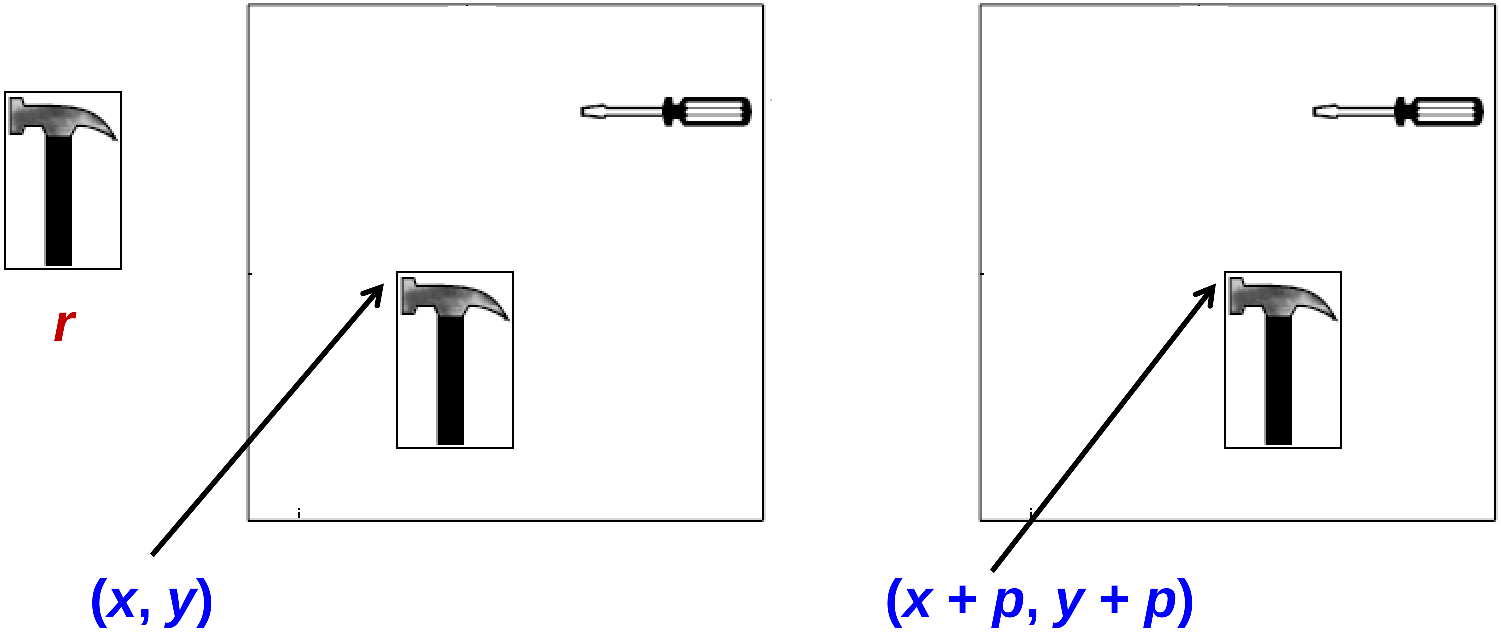
Computation Considerations in Correlation Based TM (2)

- Limit the search space
  - Search only in the area of $[-p, p]$ X $[-p, p]$ centered at (x, y)



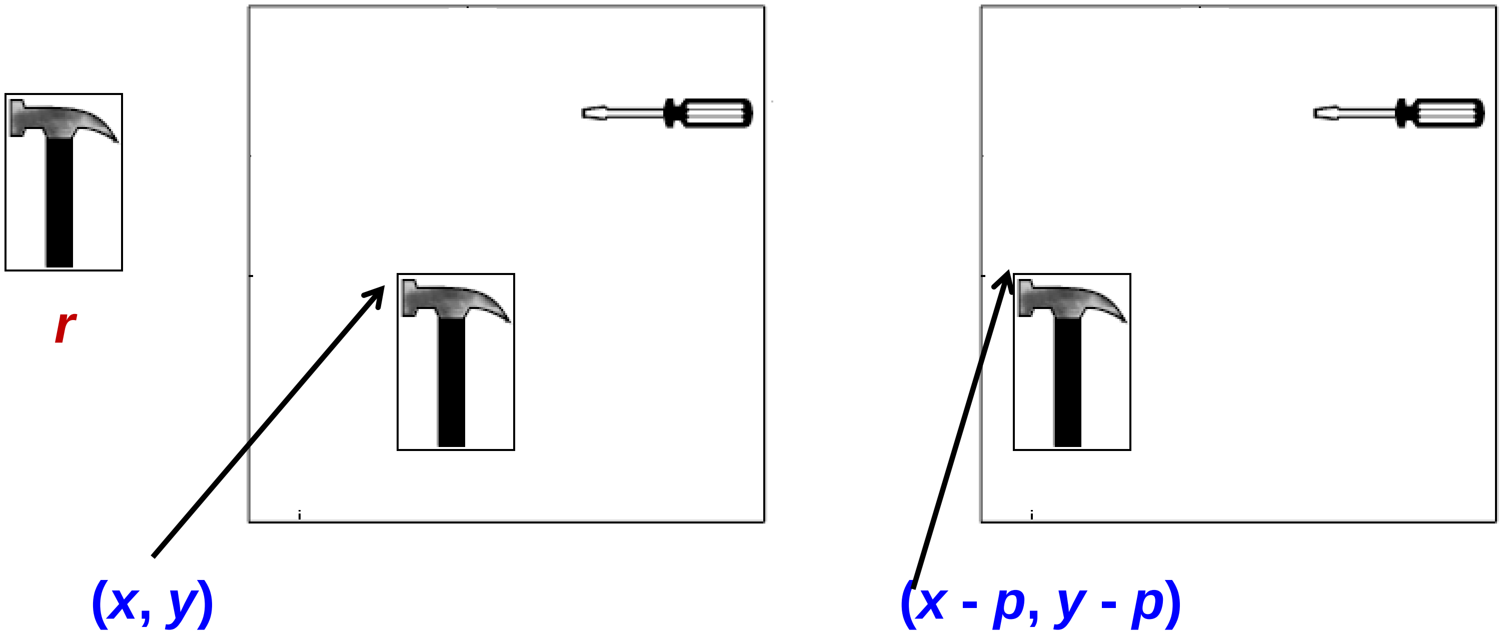
Computation Considerations in Correlation Based TM (2)

- Limit the search space
  - Search only in the area of [-*p*, *p*] X [-*p*, *p*] centered at (*x*, *y*)



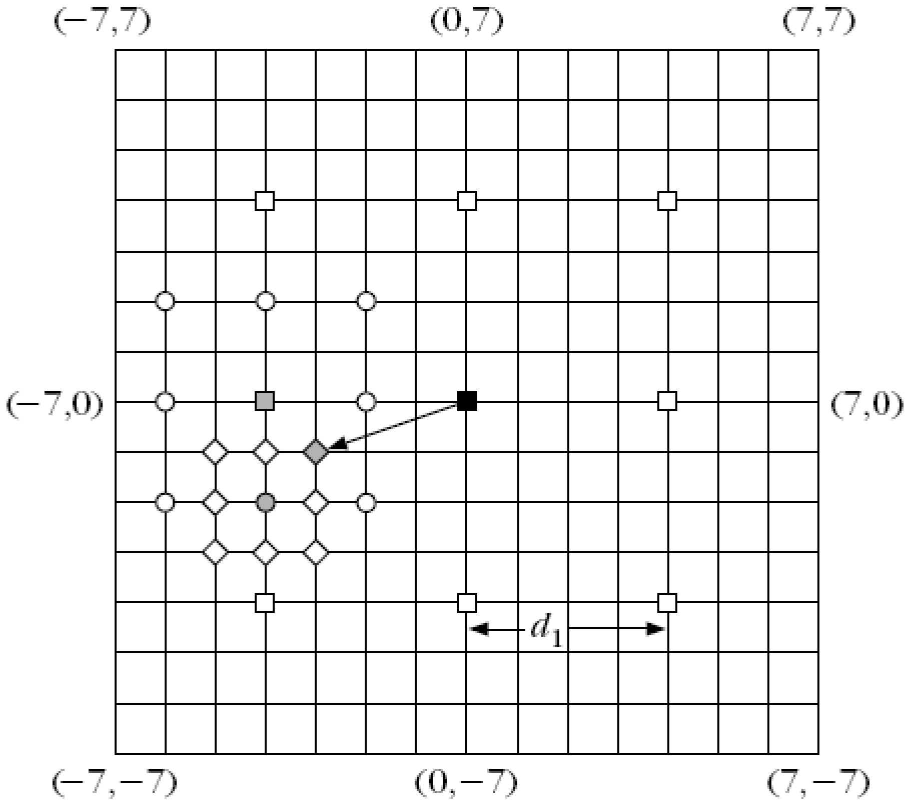
Computation Considerations in Correlation Based TM (2)

- Limit the search space
  - Search only in the area of $[-p, p] \times [-p, p]$ centered at (x, y)



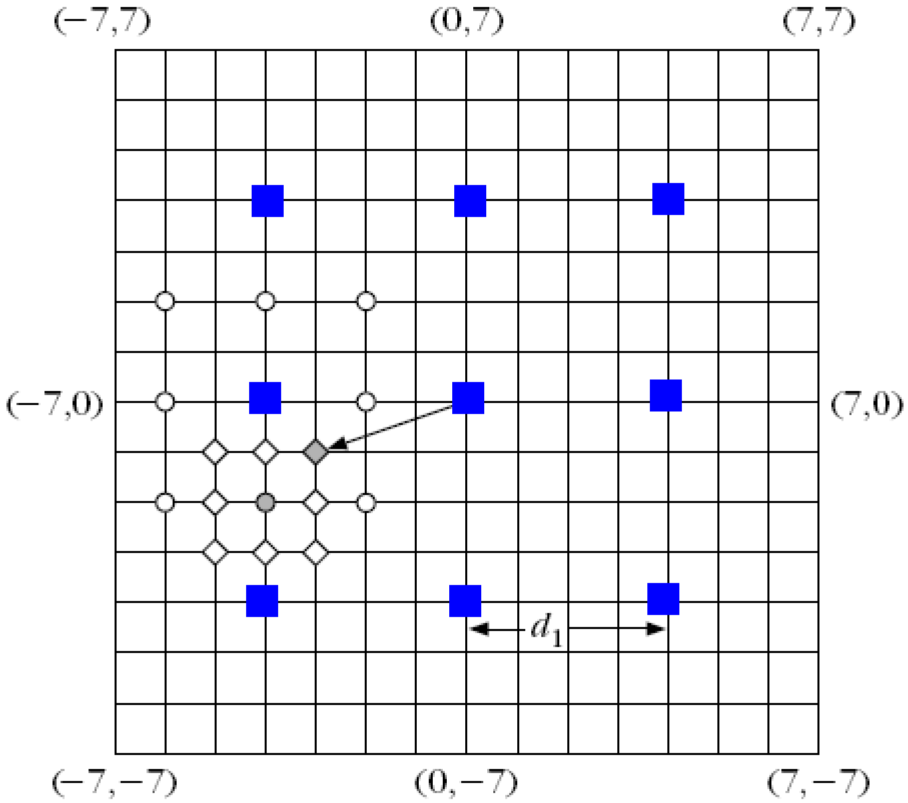
Computation Considerations in Correlation Based TM (3)

- 2D Logarithmic search
  - Start with a rectangle of size [-*p*, *p*] X [-*p*, *p*]



Computation Considerations in Correlation Based TM (3)

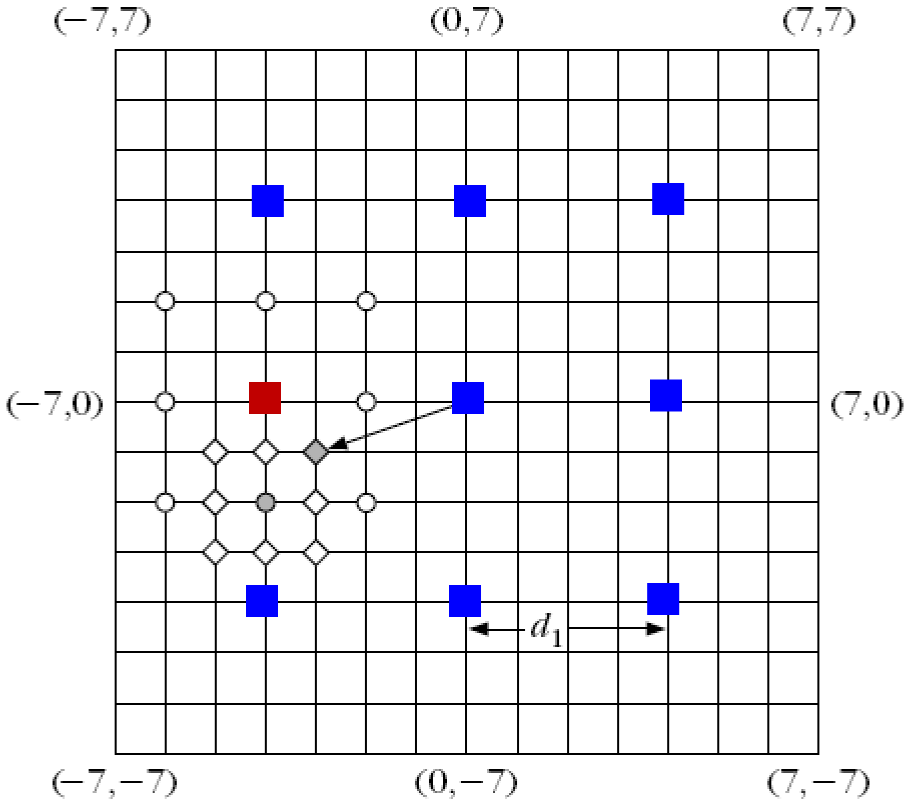
- 2D Logarithmic search
  - Search only at 9 points separated by d_1



$$d_1 = 2^{k-1}$$
$$k = \lceil \log_2 p \rceil$$

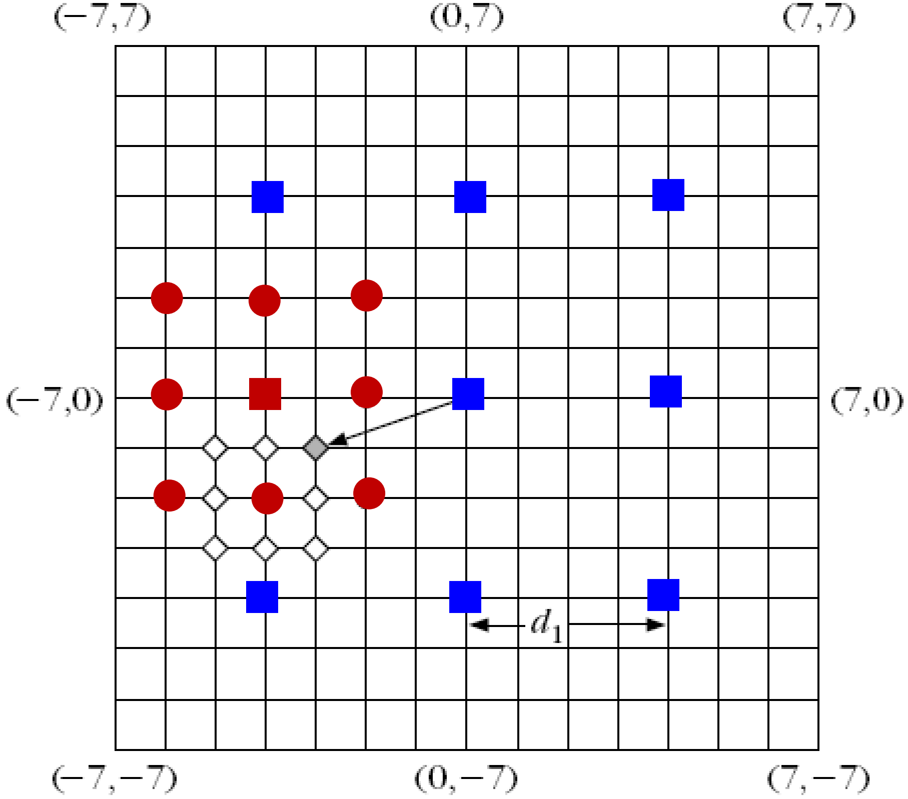
Computation Considerations in Correlation Based TM (3)

- 2D Logarithmic search
  - Maximum found at dark red



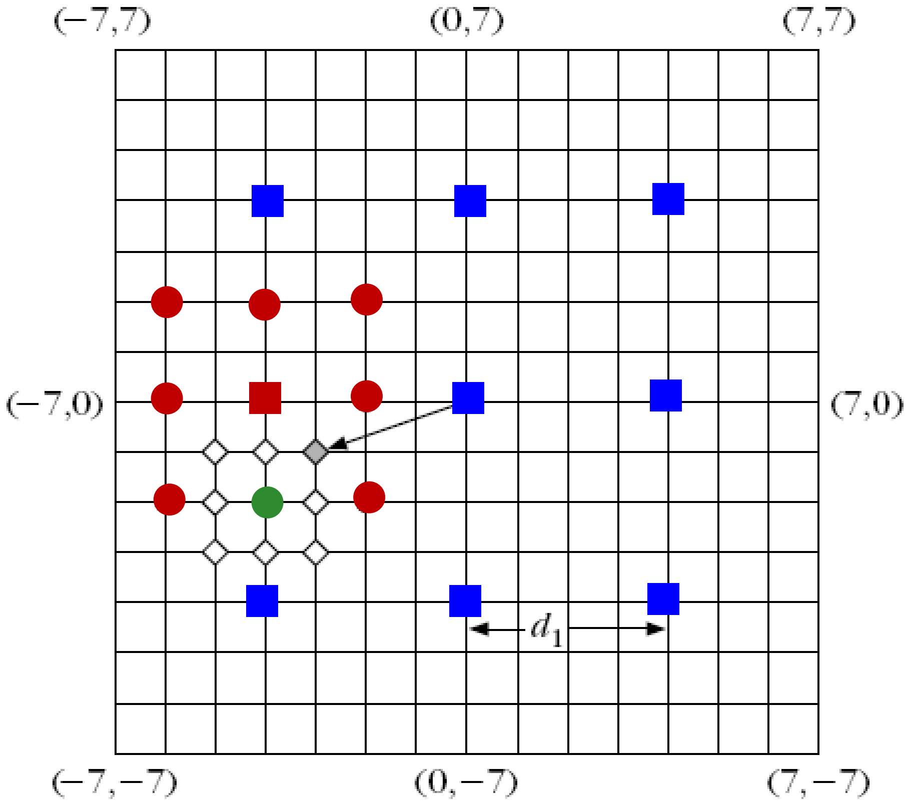
Computation Considerations in Correlation Based TM (3)

- 2D Logarithmic search
  - Search in the rectangle of size [-*p/4*, *p/4*] X [-*p/4*, *p/4*] centered at dark red



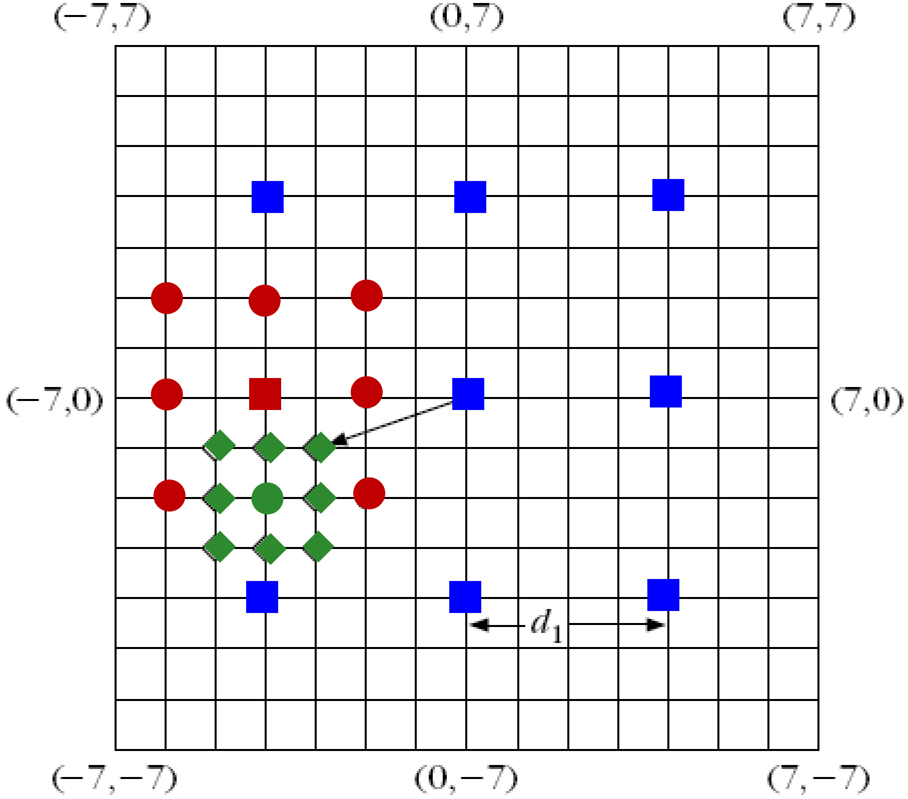
Computation Considerations in Correlation Based TM (3)

- 2D Logarithmic search
  - Maximum found at green



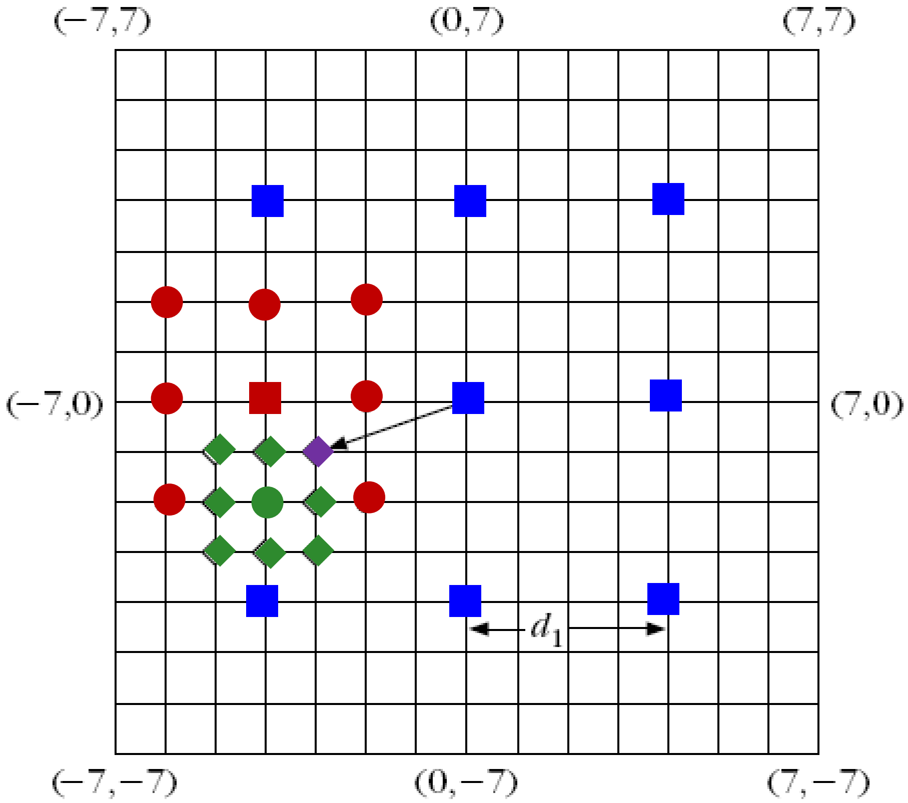
Computation Considerations in Correlation Based TM (3)

- 2D Logarithmic search
  - Search in the rectangle of size [-*p/8*, *p/8*] X [-*p/8*, *p/8*] centered at green



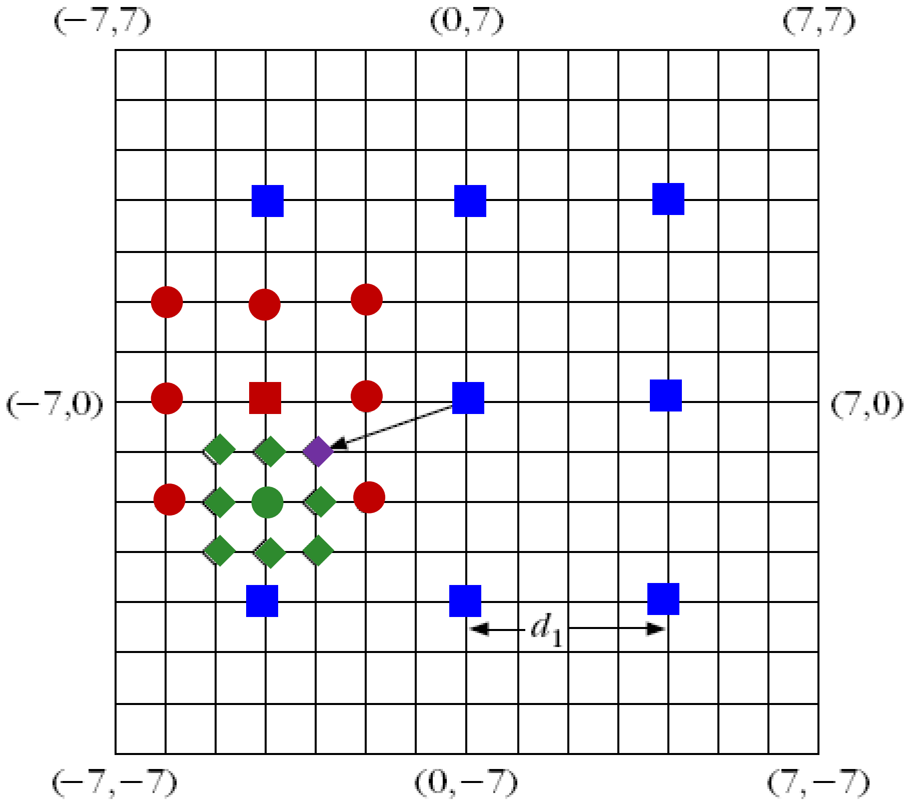
Computation Considerations in Correlation Based TM (3)

- 2D Logarithmic search
  - Maximum found at purple



Computation Considerations in Correlation Based TM (3)

- Complexity $MN(8k+1)$



$$k = \lceil \log_2 p \rceil$$

Computation Considerations in Correlation Based TM (4)

- Hierarchical Search
  - Search the reference in the area of size [-*p*, *p*] X [-*p*, *p*] centered at (*x*, *y*)
  - Let, reference be of size 16X16

reference

test

Computation Considerations in Correlation Based TM (4)

- Hierarchical Search

Level 0

Original reference and test image

Low pass Filter of Level 0

Computation Considerations in Correlation Based TM (4)

- Hierarchical Search

Level 0

Low pass Filter of Level 0

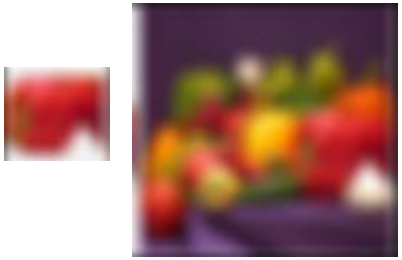
Sub-sampled by 2

Level 1

Computation Considerations in Correlation Based TM (4)

- Hierarchical Search

Level 1



Low pass Filter of Level 1

Sub-sampled by 2

Level 2

Computation Considerations in Correlation Based TM (4)

- Hierarchical Search

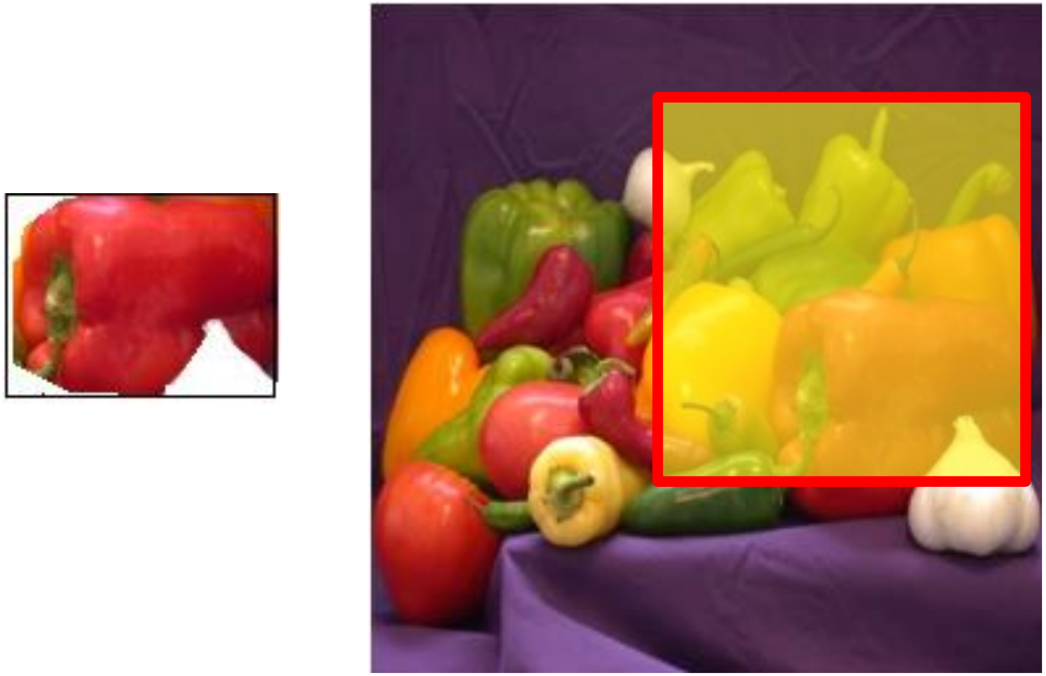
Level 0

Level 1

Level 2

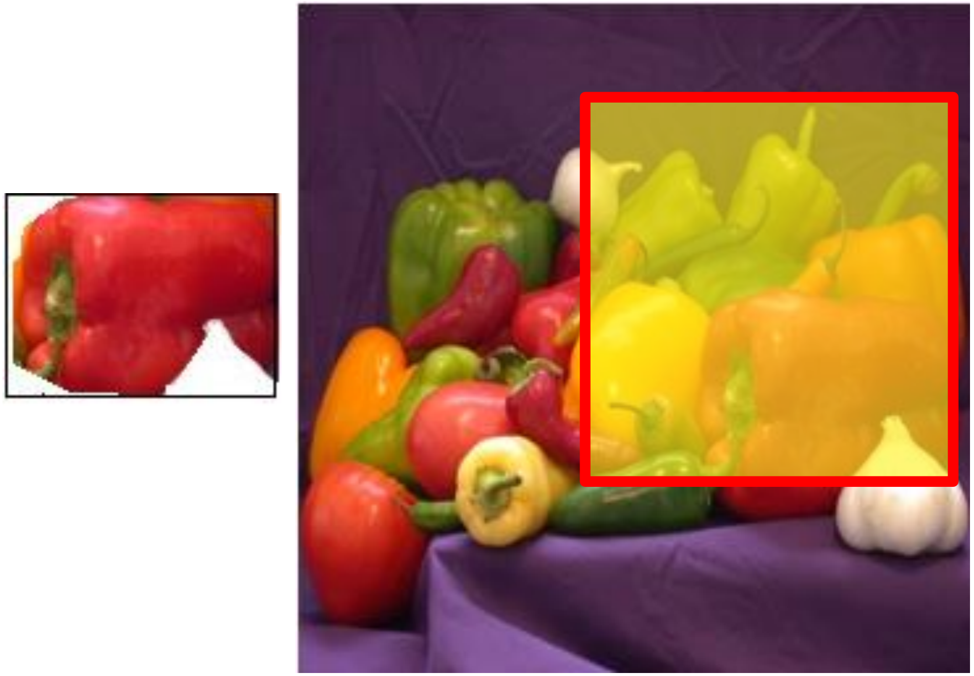
Computation Considerations in Correlation Based TM (4)

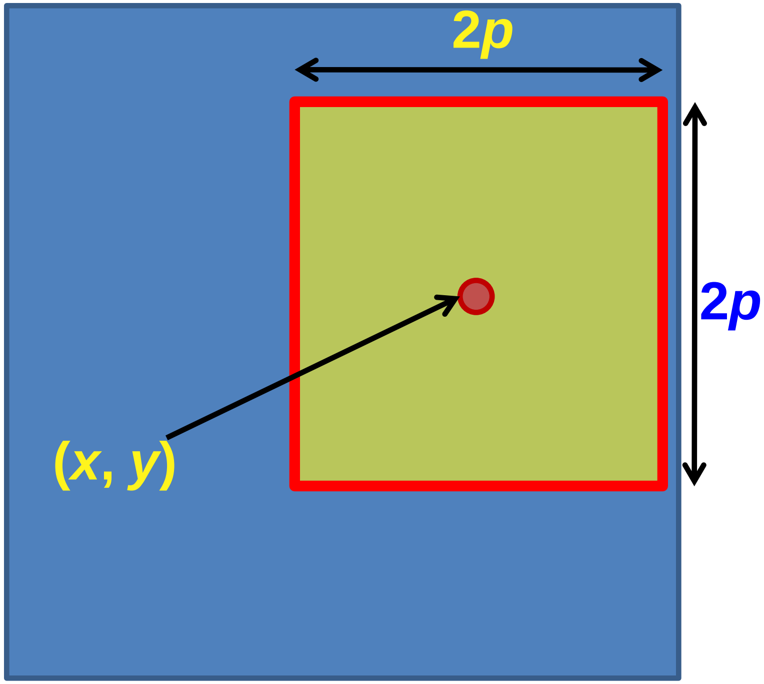
- Hierarchical Search



Computation Considerations in Correlation Based TM (4)

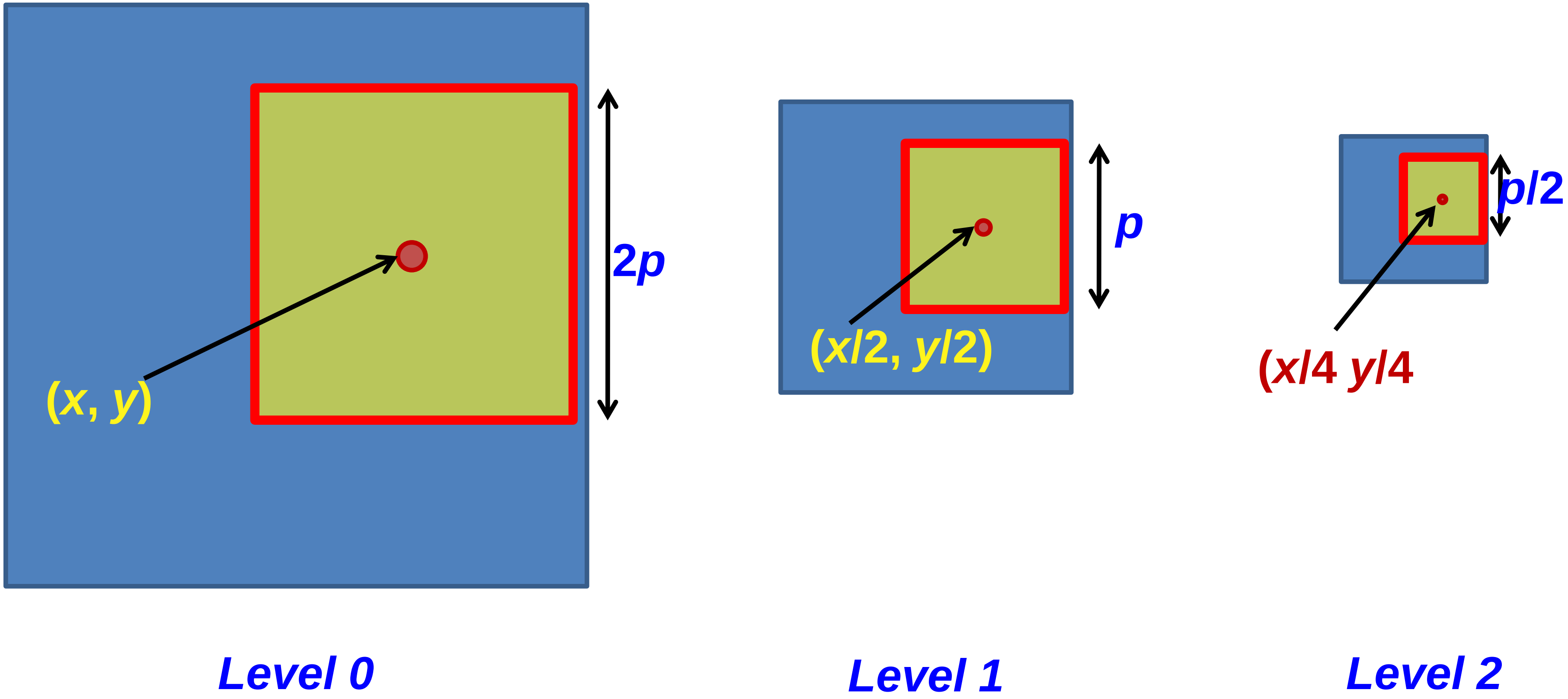
- Hierarchical Search





Computation Considerations in Correlation Based TM (4)

- Hierarchical Search



Computation Considerations in Correlation Based TM (4)

- Hierarchical Search
  - Start at Level 2 with the reference of size 4X4
  - Search in the rectangle $[-p/4, p/4]$ $[-p/4, p/4]$ centered at $(x/4, y/4)$

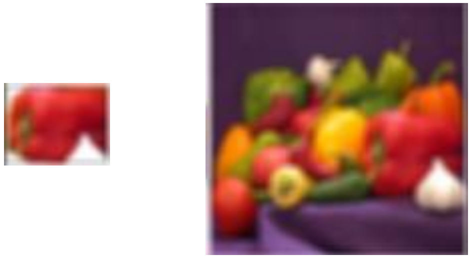
Computation Considerations in Correlation Based TM (4)

- Hierarchical Search
  - Start at Level 2 with the reference of size 4X4
  - Search in the rectangle $[-p/4, p/4]$ $[-p/4, p/4]$ centered at $(x/4, y/4)$

  - Let optimal found at (x_1, y_1) *with respect to* $(x/4, y/4)$.

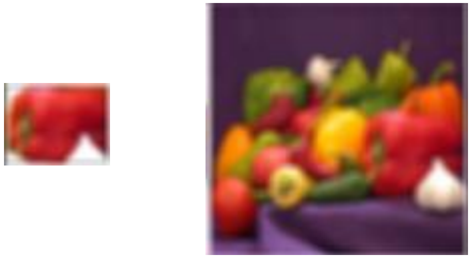
Computation Considerations in Correlation Based TM (4)

- Hierarchical Search
  - At Level 1, with the reference of size 8X8
  - Search in the rectangle [-1, 1] X [-1, 1] centered at $(x/2 + 2x_1, y/2+2y_1)$



Computation Considerations in Correlation Based TM (4)

- Hierarchical Search
  - At Level 1, with the reference of size 8X8
  - Search in the rectangle [-1, 1] X [-1, 1] centered at $(x/2 + 2x_1, y/2 + 2y_1)$



  - Let optimal found at (x_2, y_2) *with respect to* $(x/2, y/2)$.

Computation Considerations in Correlation Based TM (4)

- Hierarchical Search
  - At Level 0, with the reference of size 16X16
  - Search in the rectangle [-1, 1] X [-1, 1] centered at $(x+2x_2, y+2y_2)$

Computation Considerations in Correlation Based TM (4)

- Hierarchical Search
  - At Level 0, with the reference of size 16X16
  - Search in the rectangle [-1, 1] X [-1, 1] centered at $(x + 2x_2, y + 2y_2)$

  - Location at this time is the final one

Computation Considerations in Correlation Based TM (4)

- Complexity of Hierarchical Search
  - 9 × No. of Decompositions +
  - Complexity at highest level