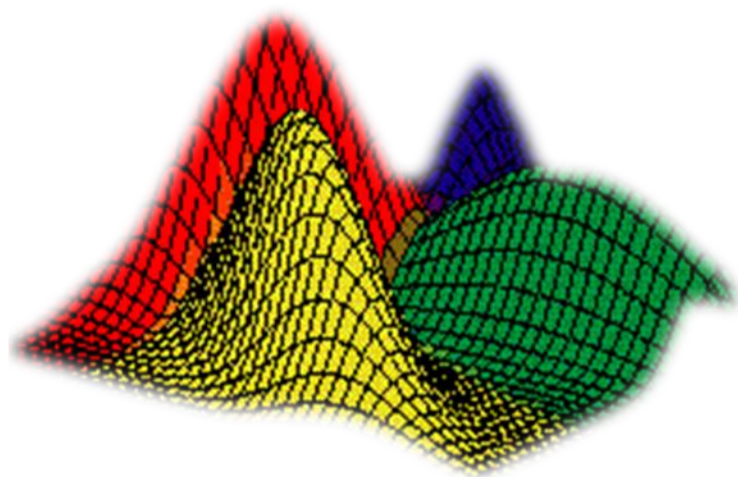
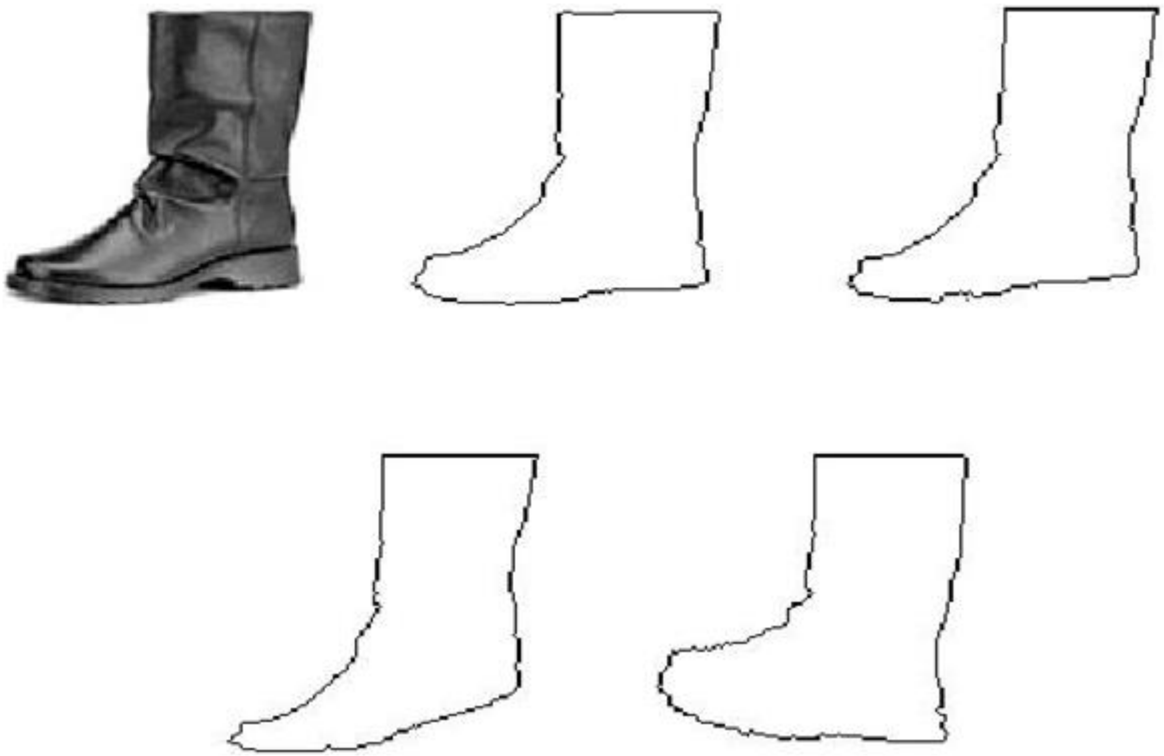


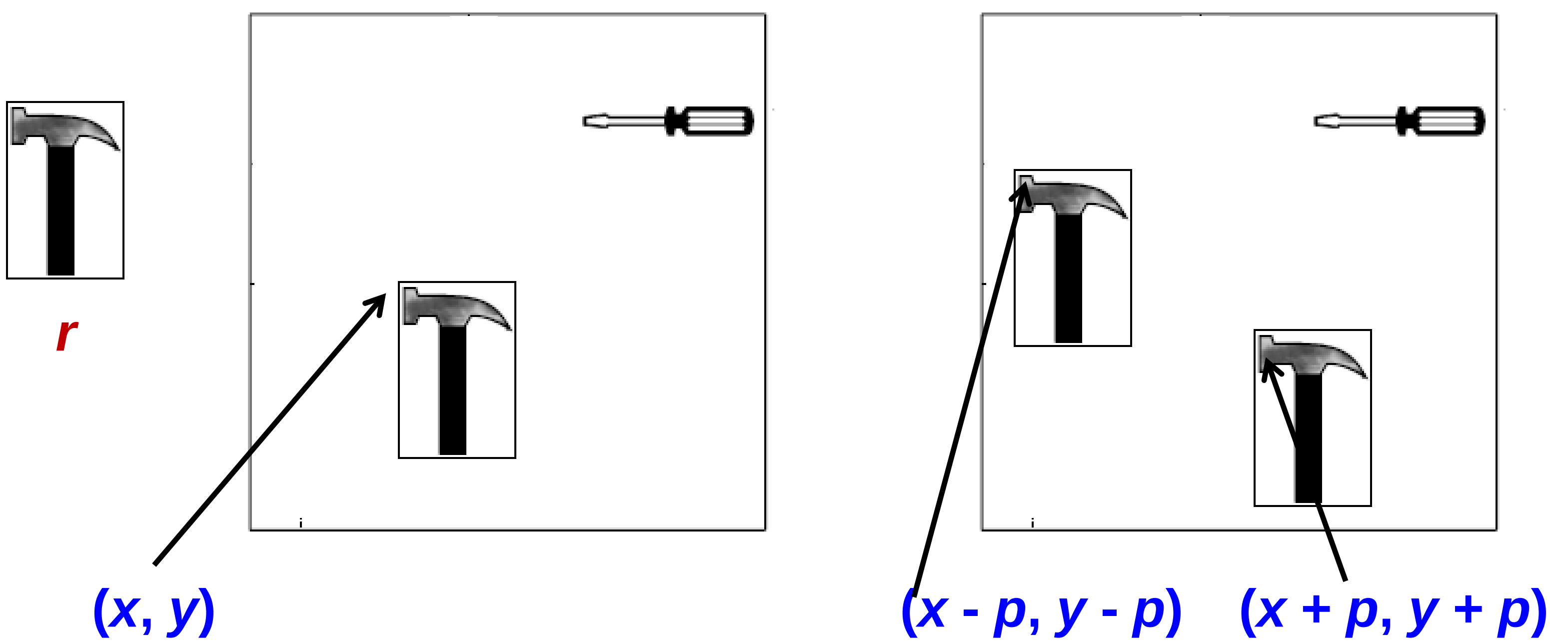
CSE 473
Pattern Recognition

Template Matching



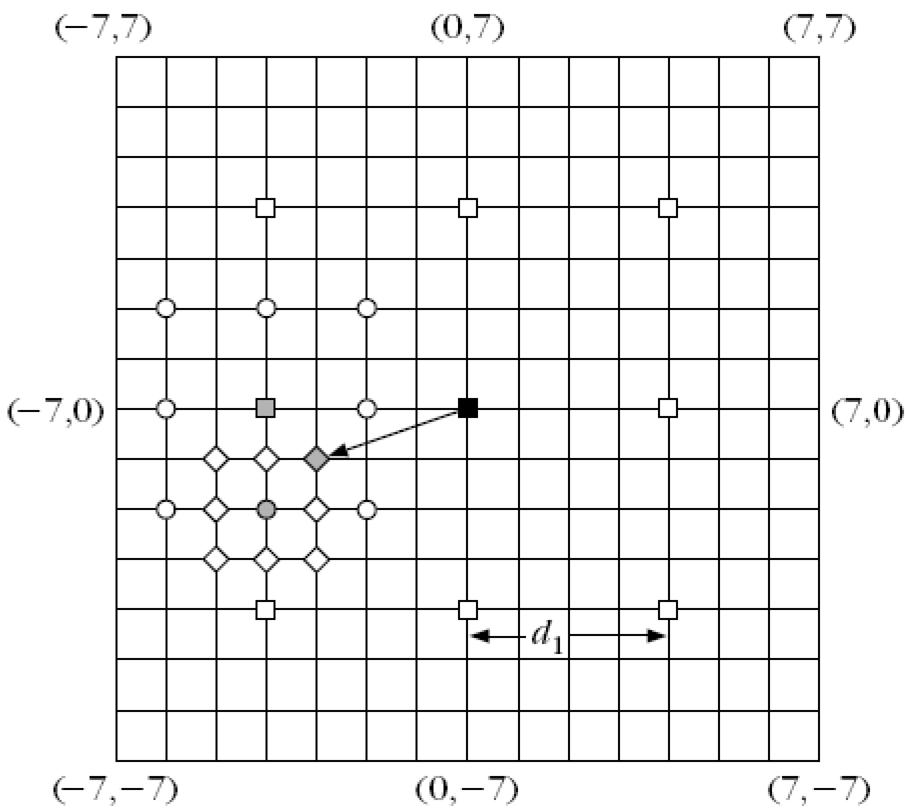
Computation Considerations in Correlation Based TM (2)

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



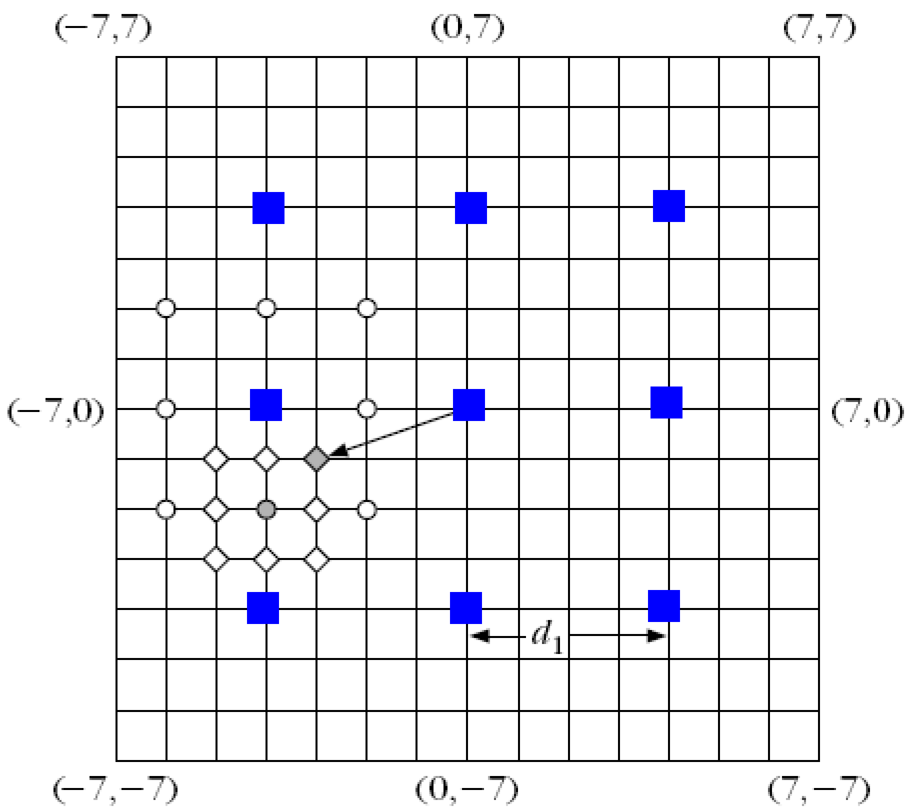
Computation Considerations in Correlation Based TM (3)

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



Computation Considerations in Correlation Based TM (3)

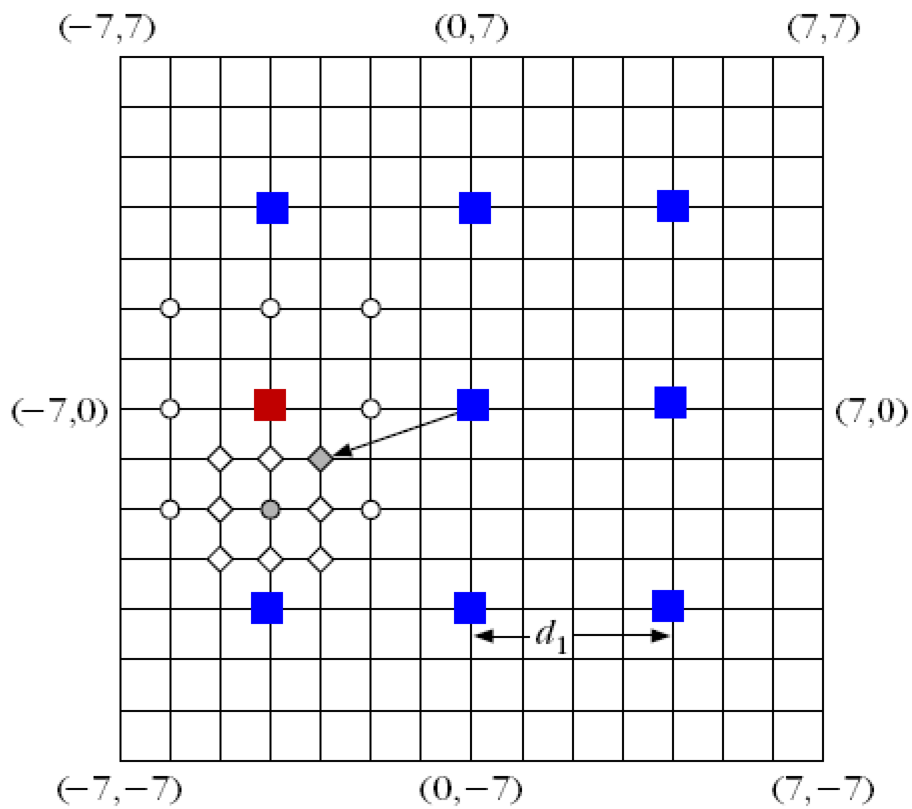
- 2D Logarithmic search
  - Search only at 9 points separated by d_1



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

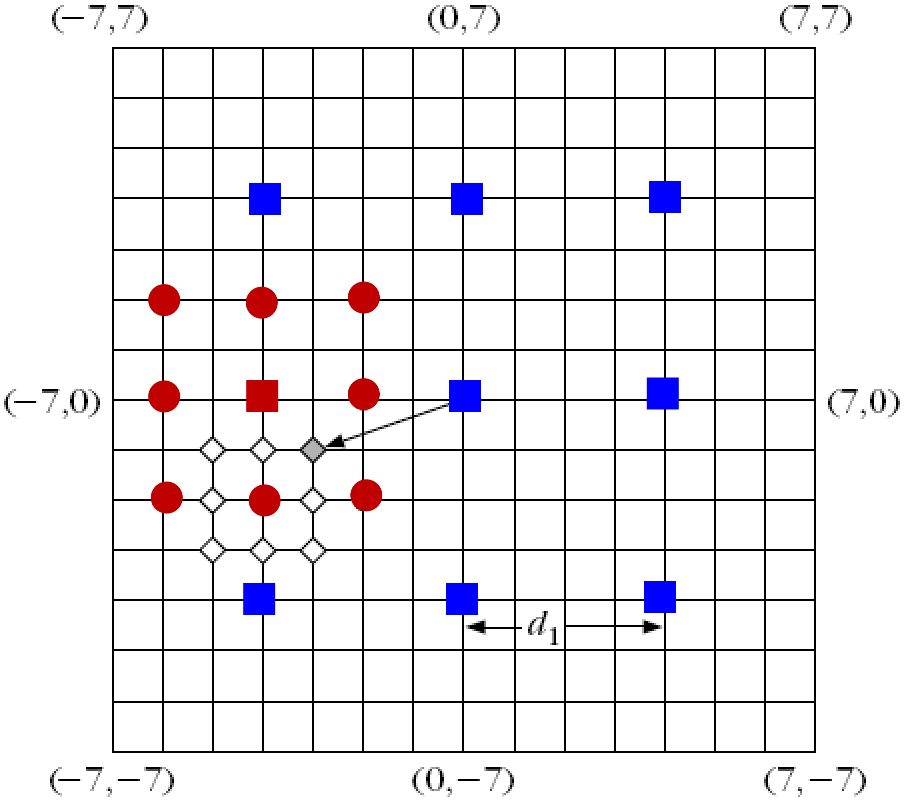
Computation Considerations in Correlation Based TM (3)

- 2D Logarithmic search
  - Maximum found at dark red



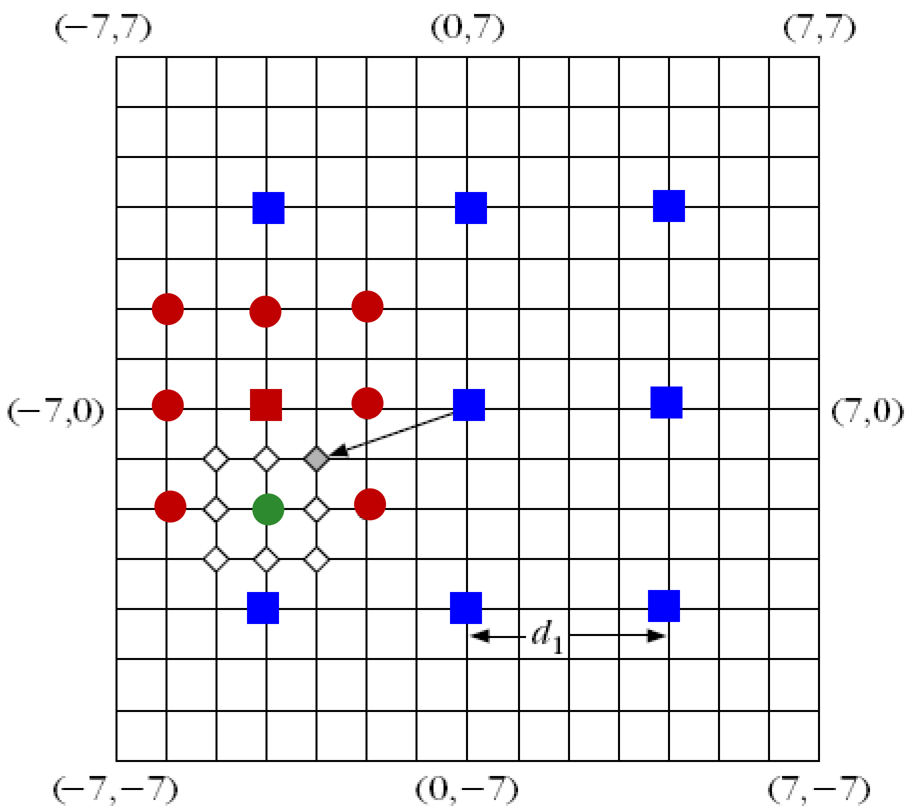
Computation Considerations in Correlation Based TM (3)

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



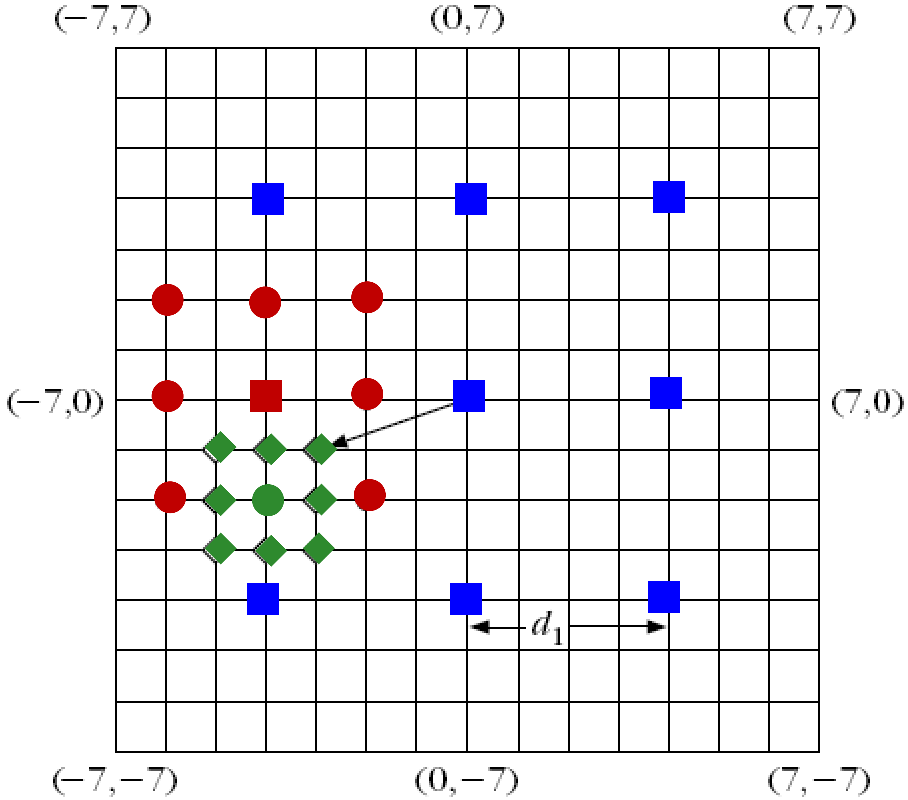
Computation Considerations in Correlation Based TM (3)

- 2D Logarithmic search
  - Maximum found at green



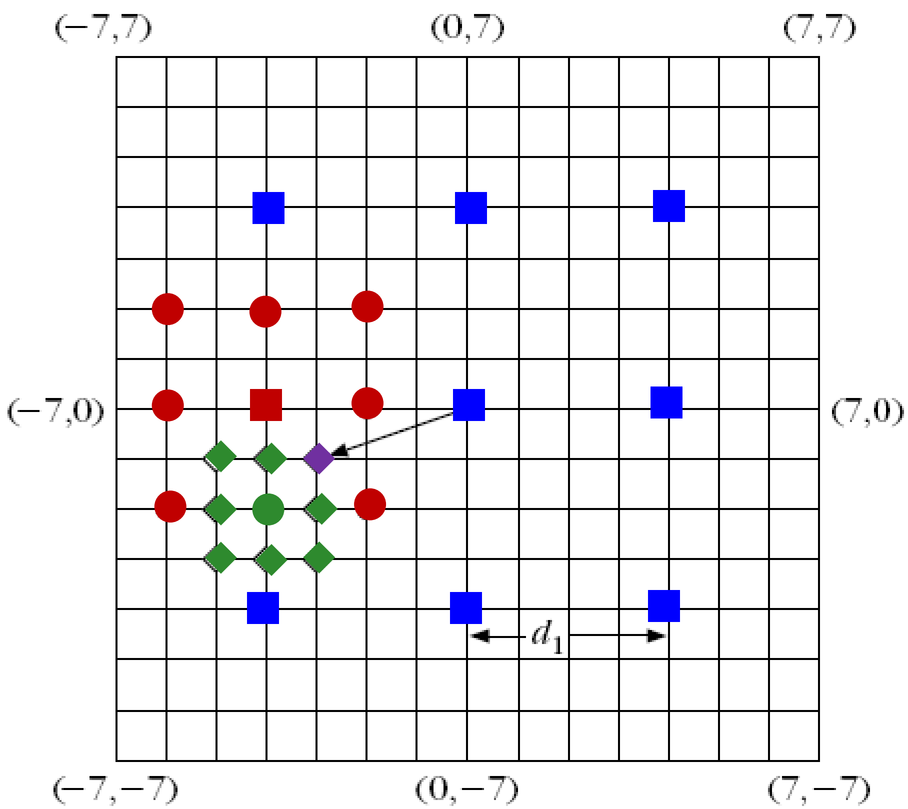
Computation Considerations in Correlation Based TM (3)

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



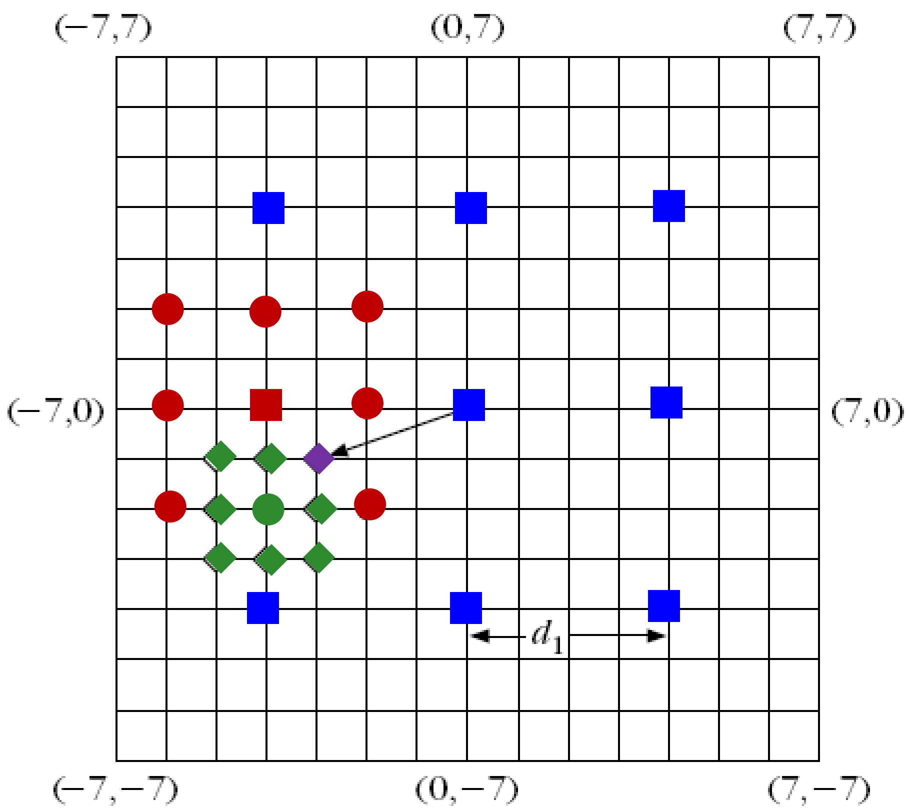
Computation Considerations in Correlation Based TM (3)

- 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

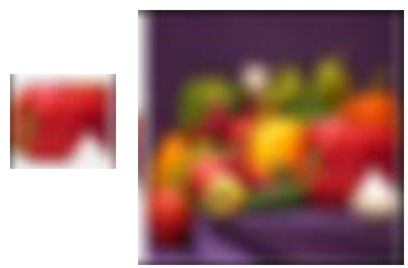
Low pass Filter of Level 0

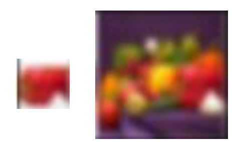
Sub-sampled by 2

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

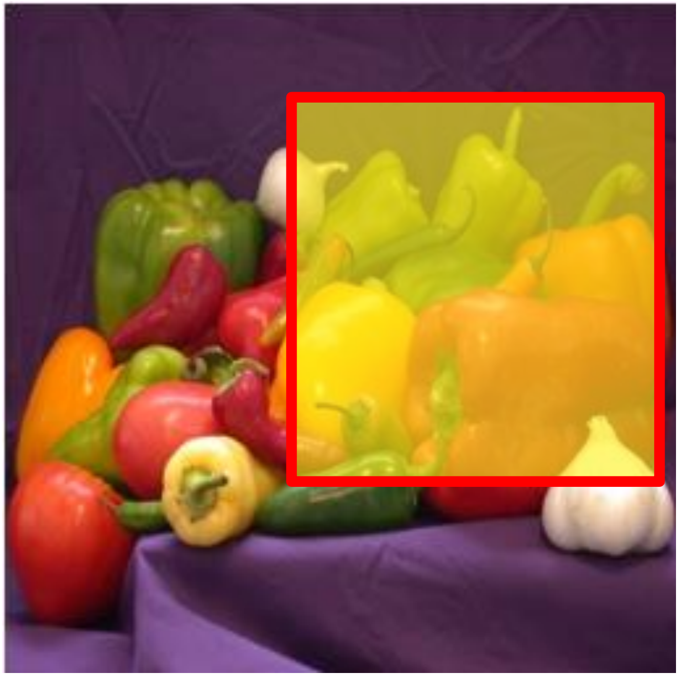
Computation Considerations in Correlation Based TM (4)

- Hierarchical Search

Level 0 *Level 1* *Level 2*

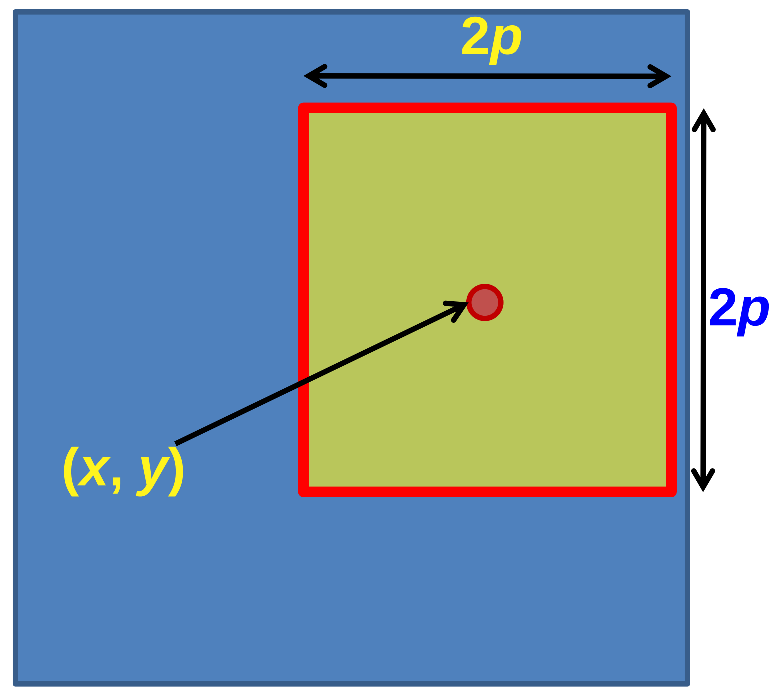
Computation Considerations in Correlation Based TM (4)

- Hierarchical Search



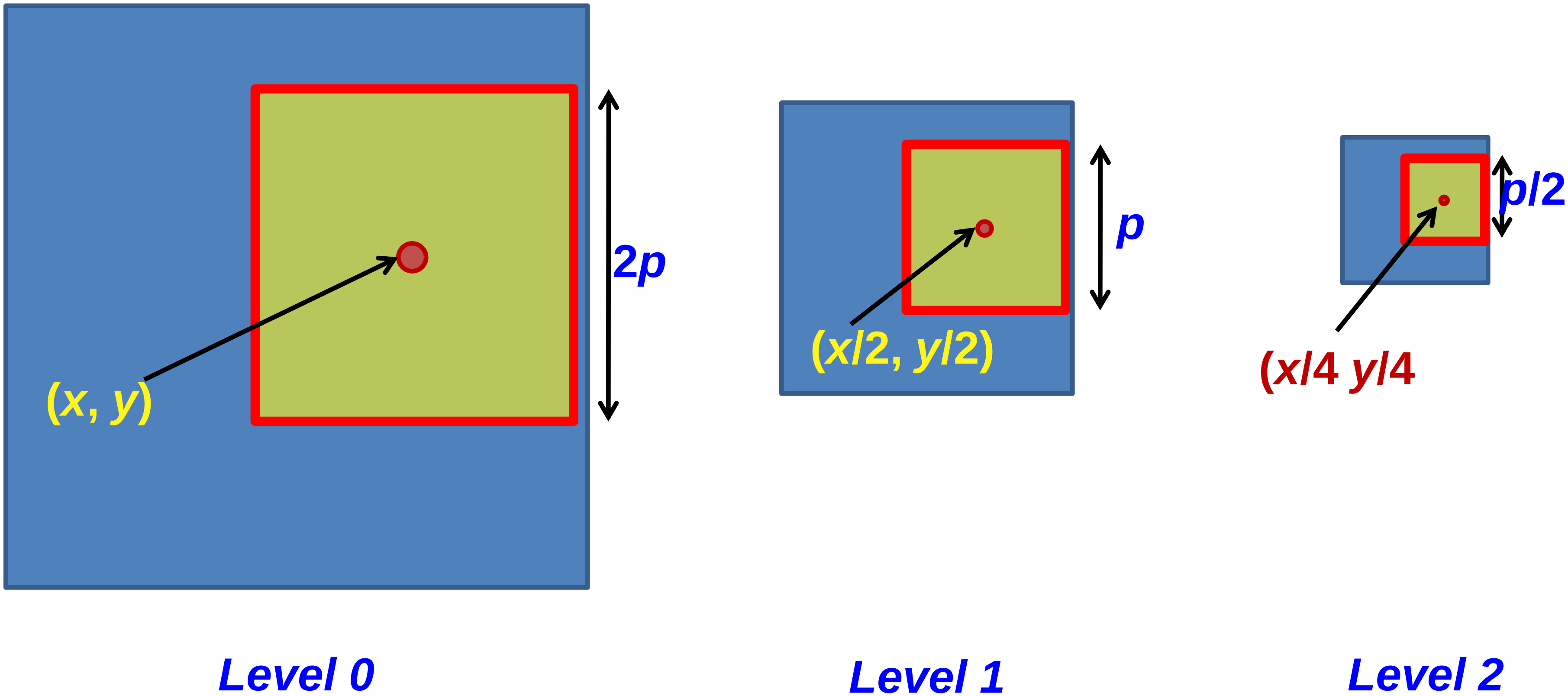
Computation Considerations in Correlation Based TM (4)

- Hierarchical Search



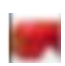
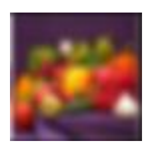
Computation Considerations in Correlation Based TM (4)

- 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 $(x+2x_2, y+2y_2)$

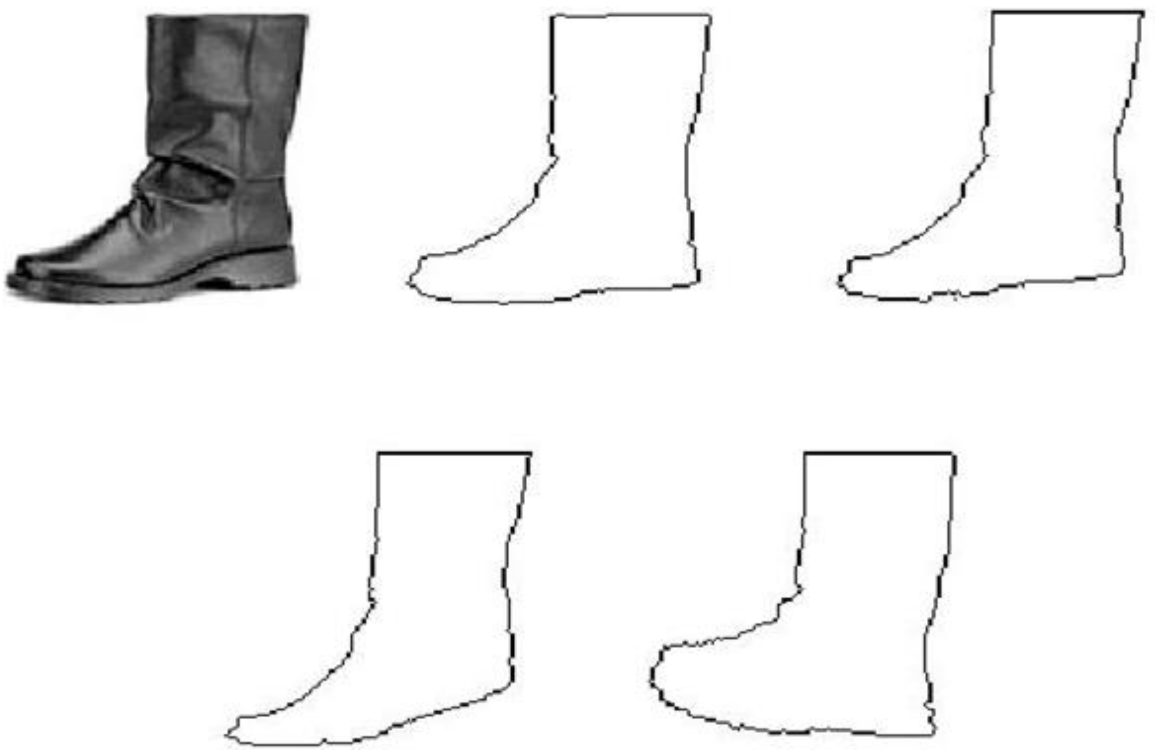

  - Location at this time is the final one

Computation Considerations in Correlation Based TM (4)

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

Deformable Template Matching

- Why is This?
  - Test and reference patterns are seldom exact
  - Rather, they are ‘similar’
  - In CBIR, query sketch significantly differs from the shapes in image DB



Deformable Template Matching

- 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 $r(i,j)$:

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

  - Match the test pattern with each of the deformed patterns

Deformable Template Matching

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

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

- The goal is to chose $\underline{\xi}$ so that $E_m\left(\underline{\xi}\right)$ is minimum

Deformable Template Matching

- However, the higher the deformation, $\underline{\xi}$ 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,

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

- Thus target: small deformation and small matching energy

Deformable Template Matching

- The essential elements
  - A prototype of the reference
  - Transformation function
  - Matching Energy cost
  - Deformation Energy cost

Deformable Template Matching

- The prototype of the reference
  - Should be representable
  - Capture the *mean shape* characteristics of an object

Deformable Template Matching

- Transformation function
  - Any appropriate parametric operation
  - A suitable transformation is:

$$(x,y) \longrightarrow (x,y) + (D^x(x,y), D^y(x,y))$$

Deformable Template Matching

- Transformation function
  - Any appropriate parametric operation
  - A suitable transformation is:

$$(x, y) \longrightarrow (x, y) + (D^x(x, y), D^y(x, y))$$

where,

$$D^x(x, y) = \sum_{m=1}^{M} \sum_{n=1}^{N} \xi^x_{mn} e^x_{mn}(x, y)$$

$$D^y(x, y) = \sum_{m=1}^{M} \sum_{n=1}^{N} \xi^y_{mn} e^y_{mn}(x, y)$$

$$e^x_{mn}(x, y) = \alpha_{mn} \sin \pi n x \cos \pi m y$$

$$e^y_{mn}(x, y) = \alpha_{mn} \cos \pi m x \sin \pi n y$$

$$\alpha_{mn} = \frac{1}{\pi^2(n^2 + m^2)}$$

Deformable Template Matching

- Deformation Energy cost
  - This should be minimum for no deformation, that is, for $\boldsymbol{\xi} = \mathbf{0}$.
  - Alternately,

$$E_d(\boldsymbol{\xi}) = \sum_m \sum_n ((\xi_{mn}^x)^2 + (\xi_{mn}^y)^2)$$

Deformable Template Matching

- Matching Energy cost
  - Captured as a function of point-to-point distance between reference and test pattern:

$$E_m(\boldsymbol{\xi}, \boldsymbol{\theta}, I) = \frac{1}{N_d} \sum_{i,j} (1 + \Phi(i, j))$$

Deformable Template Matching

- Matching Energy cost
  - Captured as a function of point-to-point distance between reference and test pattern:

$$E_m(\boldsymbol{\xi}, \boldsymbol{\theta}, I) = \frac{1}{N_d} \sum_{i,j} (1 + \Phi(i, j))$$

where,

$$\Phi(i,j) = -\exp\left(-\rho(\delta_i^2 + \delta_j^2)^{1/2}\right)$$

- (δ_i, δ_j) is the displacement of the (*i*, *j*) pixel of the deformed template from the nearest pixel of the test template
- ρ is a constant

Context Dependent Classification

Context Dependent Classification

- Recall context free classification
  - No relation exist among classes
  - No relation exists among objects (feature vectors)
  - A new object is classified to any class independent of the previous objects' classes

Context Dependent Classification

- In Context dependent classification, the class of a feature vector depends on
  - Its own value
  - Value of other feature vectors
  - Classes assigned to other vectors

Context Dependent Classification

- Application
  - Communication
  - Image Processing
  - Signal Processing

Solution for Context Dependent Classification

- Recall Bayesian formulation for context free classification
  - Assign x to ω_i if $P(\omega_i|\underline{x}) > P(\omega_j|\underline{x}),\ \forall j \neq i$
- In context dependent classification, we cannot apply it directly because of interdependency of features and classes

Solution for Context Dependent Classification

- This interrelation demands the classification to be performed simultaneously for all available feature vectors

- we assume that the training vectors $\underline{x}_1, \underline{x}_2, \ldots, \underline{x}_N$ occur in sequence, one after the other and we will refer to them as **observations**

Context Dependent Bayesian Classifier

- Let $X:\{\underline{x}_1, \underline{x}_2, \ldots, \underline{x}_N\}$ be sequence of observations

- Let $\omega_i,\ i = 1,2,\ldots,M$ be the available *M* classes

- Let Ω_i be a possible sequence of assigned classes, that is

$$\Omega_i : \omega_{i1}\ \omega_{i2} \ldots \omega_{iN}$$

where, $i_k \in \{1, 2, \ldots, M\}$ for $k = 1, 2, \ldots, N$

- There are M^N of Ω_i

Context Dependent Bayesian Classifier

- Now, given $X : \{\underline{x}_1, \underline{x}_2, \dots, \underline{x}_N\}$ and $\Omega_i : \omega_{i1}\, \omega_{i2} \dots \omega_{iN}$

  Classify X to using the Bayesian rule

$$\boxed{X \rightarrow \Omega_i : \quad P(\Omega_i | X) > P(\Omega_j | X) \ \ \forall i \neq j, \quad i, j = 1,2,\dots, M^N}$$

- This is equivalent to classifying

$$\boldsymbol{x}_1 \text{ to class } \omega_{i_1}, \boldsymbol{x}_2 \text{ to } \omega_{i_2}, \text{and so on}$$

Context Dependent Bayesian Classifier

- The rule

$$\boxed{P(\Omega_i|X) > P(\Omega_j|X) \;\; \forall i \neq j}$$

can be simplified as

$$P(\Omega_i)p(X|\Omega_i) > P(\Omega_j)p(X|\Omega_j), \quad \forall i \neq j$$

Further Simplification: Markov Chain Model

$$P(\Omega_i)p(X|\Omega_i) > P(\Omega_j)p(X|\Omega_j), \quad \forall i \neq j$$

- Markov Chain Models (for class dependence)

$$P(\omega_{i_k} \middle| \omega_{i_{k-1}}, \omega_{i_{k-2}}, \ldots, \omega_{i_1}) = P(\omega_{i_k} \middle| \omega_{i_{k-1}})$$

which means class dependence is limited to only within two successive classes

Further Simplification: Markov Chain Model

- Markov Chain Models (for class dependence)

$$P(\omega_{i_k} | \omega_{i_{k-1}}, \omega_{i_{k-2}}, \ldots, \omega_{i_1}) = P(\omega_{i_k} | \omega_{i_{k-1}})$$

in other words,

if observations $x_{k-1}, x_{k-2}, \ldots, x_1$ belong to classes $\omega_{i_{k-1}}, \omega_{i_{k-2}}, \ldots, \omega_{i_1}$

then observation x_k, at stage k, belonging to class ω_{i_k}

depends on the *class from which observation* $\boldsymbol{x}_{k-1}$*, at stage k*-1 *has occurred*

Further Simplification: Markov Chain Model

- Markov Chain Models (for class dependence)

$$P(\omega_{i_k} | \omega_{i_{k-1}}, \omega_{i_{k-2}}, \ldots, \omega_{i_1}) = P(\omega_{i_k} | \omega_{i_{k-1}})$$

- Therefore, we can write

$$P(\Omega_i) \equiv P(\omega_{i_1}, \omega_{i_2}, \ldots, \omega_{i_N})$$

$$= P(\omega_{i_N} | \omega_{i_{N-1}}, \ldots, \omega_{i_1}) P(\omega_{i_{N-1}} | \omega_{i_{N-2}}, \ldots, \omega_{i_1}) \ldots P(\omega_{i_1})$$

Further Simplification: Markov Chain Model

- Markov Chain Models (for class dependence)

$$P(\omega_{i_k} \mid \omega_{i_{k-1}}, \omega_{i_{k-2}}, \ldots, \omega_{i_1}) = P(\omega_{i_k} \mid \omega_{i_{k-1}})$$

- Therefore, we can write

$$P(\Omega_i) \equiv P(\omega_{i_1}, \omega_{i_2}, \ldots, \omega_{i_N})$$
$$= P(\omega_{i_N} \mid \omega_{i_{N-1}}, \ldots, \omega_{i_1}) P(\omega_{i_{N-1}} \mid \omega_{i_{N-2}}, \ldots, \omega_{i_1}) \ldots P(\omega_{i_1})$$

- We find,

$$P(\Omega_i) = P(\omega_{i_1}) \prod_{k=2}^{N} P(\omega_{i_k} \mid \omega_{i_{k-1}})$$

Further Simplification: Markov Chain Model

- Further assumption
  - $\underline{x}_i$ statistically mutually independent

$$
\boxed{
\begin{aligned}
p(X|\Omega_i) &= p(\underline{x}_1, \underline{x}_2, \underline{x}_3 \cdots, \underline{x}_N|\Omega_i) \\
&= p(\underline{x}_1|\Omega_i)\, p(\underline{x}_2|\Omega_i)\, p(\underline{x}_3|\Omega_i) \cdots p(\underline{x}_N|\Omega_i) \\
&= \prod_{k=1}^{N} p(\underline{x}_k|\Omega_i)
\end{aligned}
}
$$

Further Simplification: Markov Chain Model

- Further assumption
  - $\underline{x}_i$ statistically mutually independent

$$\boxed{\begin{aligned} p(X|\Omega_i) &= p(\underline{x}_1, \underline{x}_2, \underline{x}_3 \cdots, \underline{x}_N|\Omega_i) \\ &= p(\underline{x}_1|\Omega_i)\, p(\underline{x}_2|\Omega_i)\, p(\underline{x}_3|\Omega_i) \cdots p(\underline{x}_N|\Omega_i) \\ &= \prod_{k=1}^{N} p(\underline{x}_k|\Omega_i) \end{aligned}}$$

  - If the pdf in one class is independent of the others, then

$$p(\vec{x}_k \mid \Omega_i) = p(\vec{x}_k \mid \omega_{i1}\, \omega_{i2} \ldots \omega_{iN}) = p(\vec{x}_k \mid \omega_{ik})$$

Further Simplification: Markov Chain Model

- Further assumption
  - $\underline{x}_i$ statistically mutually independent

$$\boxed{\begin{aligned} p(X|\Omega_i) &= p(\underline{x}_1, \underline{x}_2, \underline{x}_3 \cdots, \underline{x}_N|\Omega_i) \\ &= p(\underline{x}_1|\Omega_i)\,p(\underline{x}_2|\Omega_i)\,p(\underline{x}_3|\Omega_i)\cdots p(\underline{x}_N|\Omega_i) \\ &= \prod_{k=1}^{N} p(\underline{x}_k|\Omega_i) \end{aligned}}$$

  - The pdf in one class independent of the others, then

$$p(\vec{x}_k \mid \Omega_i) = p(\vec{x}_k \mid \omega_{i1}\ \omega_{i2} \ldots \omega_{iN}) = p(\vec{x}_k \mid \omega_{ik})$$

Finally,

$$\boxed{p(X|\Omega_i) = \prod_{k=1}^{N} p(\underline{x}_k|\omega_{i_k})}$$

Further Simplification: Markov Chain Model

- From the above, the Bayes rule is readily seen to be equivalent to:

$$P(\Omega_i|X) \;\; (><) \;\; P(\Omega_j|X)$$

$$P(\Omega_i)p(X|\Omega_i) \;\; (><) \;\; P(\Omega_j)p(X|\Omega_j)$$

that is, it rests on

$$\boxed{\begin{array}{l} p(X|\Omega_i)P(\Omega_i) = P(\omega_{i_1})p(\underline{x}_1|\omega_{i_1}). \\ \prod_{k=2}^{N} P(\omega_{i_k}|\omega_{i_{k-1}})p(\underline{x}_k|\omega_{i_k}) \end{array}}$$

- To find the above maximum in brute-force task we need $O(NM^N)$ operations!!