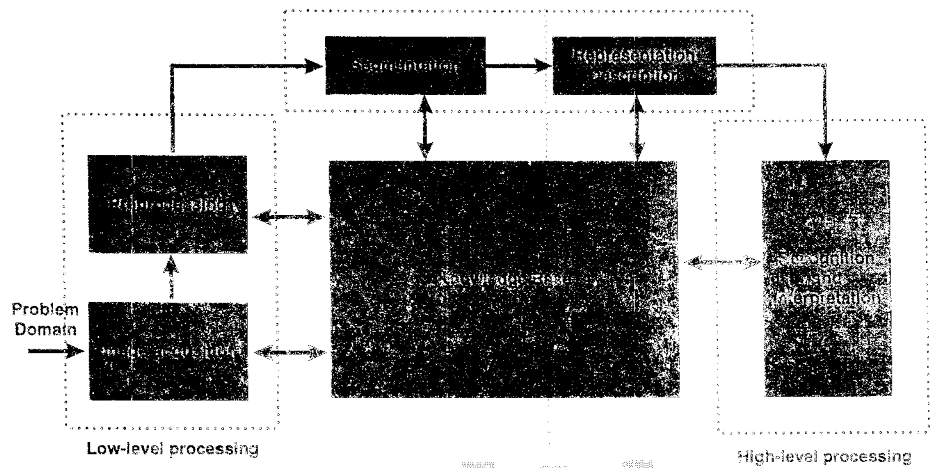
영상분석 Ⅱ : Image Analysis

김 재 석
삼성종합기술원

Contents

I. INTRODUCTION

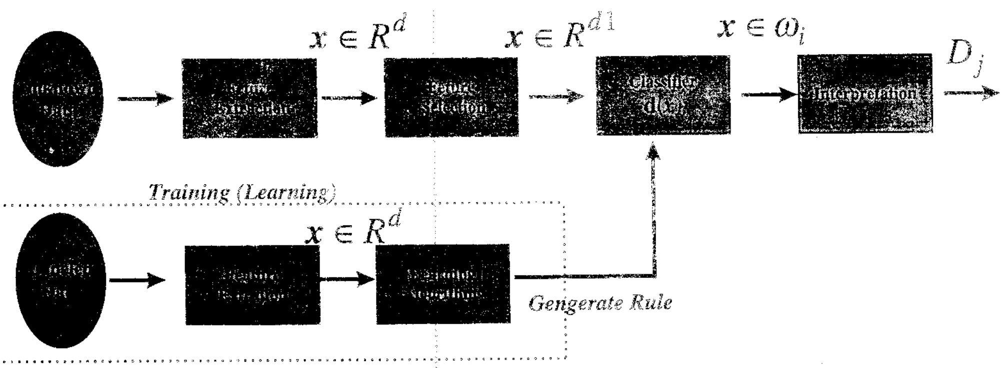
Elements of Image Analysis



- Low-level processing deals with functions that may be viewed as automatic reactions, requiring no intelligence on the part of the image analysis system(Reduce Noise , Deblurring).
- Intermediate-level processing deals with the task of extracting and characterizing components(regions) in an image resulting from a low-level process.
- High-level processing involves recognition and interpretation. These two processes have a stronger resemblance of what generally meant by the term intelligent cognition.

II. PATTERN RECOGNITION

Elements of Pattern Recognition



III. DECISION-THEORETIC METHODS

Let $\boldsymbol{x}=(x_1, x_2, ..., x_n)^T$ represent an n-dimensional pattern vector. For M pattern classes ω_1, ω_2, ..., ω_M , decision functions $d_1(\boldsymbol{x})$, $d_2(\boldsymbol{x})$, ..., $d_M(\boldsymbol{x})$ with the property that, if a pattern $\boldsymbol{x}$ belongs to class ω_i, then

$$d_i(x) > d_j(x) \qquad j = 1, 2, ..., M; \ j \neq i$$

The decision boundary separating class ω_i from ω_j given by values of $\boldsymbol{x}$ for which $d_i(\boldsymbol{x}) = d_j(\boldsymbol{x})$ or, equivalently, by values of $\boldsymbol{x}$ for which

$$d_i(\boldsymbol{x}) - d_j(\boldsymbol{x}) = 0$$

1. Minimum distance classifier

Prototype vector of class ω_j:

$$\boldsymbol{m}_j = \frac{1}{N_i} \sum_{\boldsymbol{x} \in \omega_j} \boldsymbol{x} \quad j = 1,2, \quad ..., M$$

Decision Criteria:

$$D_j(\boldsymbol{x}) = \left\| \boldsymbol{x} - \boldsymbol{m}_j \right\| \quad j = 1,2,...., M$$

$$d_j(\boldsymbol{x}) = \boldsymbol{x}^T \boldsymbol{m}_j - \frac{1}{2} \boldsymbol{m}_j^T \boldsymbol{m}_j \qquad j = 1,2,...., M$$

Decision Boundary:

$$d_{ij}(\boldsymbol{x}) = d_i(\boldsymbol{x}) - d_j(\boldsymbol{x})$$

$$= \boldsymbol{x}^T (\boldsymbol{m}_i - \boldsymbol{m}_j) - \frac{1}{2} (\boldsymbol{m}_i - \boldsymbol{m}_j)^T (\boldsymbol{m}_i - \boldsymbol{m}_j) = 0$$

Known as a hyperplane.

2. Optimal Statistical Classifiers

The probability that a particular pattern $\boldsymbol{x}$ comes from class ω_i is denoted $p(\omega_i/\boldsymbol{x})$. If the pattern classifier decides that $\boldsymbol{x}$ came from ω_j when it actually came from ω_i, it incurs a loss, denoted L_{ij}. As pattern $\boldsymbol{x}$ may belong to any one of M classes under consideration, the average loss incurred in assigning $\boldsymbol{x}$ to class ω_j is

Conditional Average risk(loss) : $r_j(\boldsymbol{x}) = \sum_{k=1}^{M} L_{kj}\, p(\omega_k / \boldsymbol{x})$

From basic probability theory, $p(a/b) = [p(a)p(b/a)]/p(b)$

$$r_j(\boldsymbol{x}) = \frac{1}{p(\boldsymbol{x})} \sum_{k=1}^{M} L_{kj}\, p(\boldsymbol{x} / \omega_k) P(\omega_k)$$

where $p(x/\omega_k)$ is the probability density function of the patterns from class ω_k and $P(\omega_k)$ is the probability of occurrence of class ω_k.

If it computes $r_1(\boldsymbol{x})$, $r_2(\boldsymbol{x})$, ..., $r_M(\boldsymbol{x})$ for each pattern $\boldsymbol{x}$, and assigns the pattern to the class with the smallest loss, the total average loss with respect to all decisions will be minimized.

__Bayes Classifier__ :

Assign an unknown pattern $\boldsymbol{x}$ to class ω_i if $r_i(\boldsymbol{x}) < r_j(\boldsymbol{x})$ for j=1, 2, ..., M; $j \neq i$.

$$\sum_{k=1}^{M} L_{ki}\, p(\boldsymbol{x} / \omega_k) P(\omega_k) < \sum_{q=1}^{M} L_{qj}\, p(\boldsymbol{x} / \omega_q) P(\omega_q)$$

Suppose $L_{ij} = 1 - \delta_{ij}$ where δ_{ij} if $i=j$ and $\delta_{ij}=0$ if $i \neq j$.

$$r_j(\boldsymbol{x}) = \sum_{k=1}^{M} (1 - \delta_{kj})\, p(\boldsymbol{x} / \omega_k) P(\omega_k)$$
$$= p(\boldsymbol{x}) - p(\boldsymbol{x} / \omega_j) P(\omega_j)$$

The Bayes classifier assigns a pattern x to class ω_i if

$$p(\boldsymbol{x}) - p(\boldsymbol{x} / \omega_i) P(\omega_i) < p(\boldsymbol{x}) - p(\boldsymbol{x} / \omega_j) P(\omega_j)$$

or

$$p(\boldsymbol{x} / \omega_i) P(\omega_i) > p(\boldsymbol{x} / \omega_j) P(\omega_j) \qquad j = 1,2,\ldots M;\ j \neq i$$

Decision function for 0-1 loss function

$$d_j(\boldsymbol{x}) = p(\boldsymbol{x} / \omega_j) P(\omega_j) \qquad j = 1,2,\ldots M$$

where a pattern vector $\boldsymbol{x}$ is assigned to class ω_i if $d_i(\boldsymbol{x}) > d_j(\boldsymbol{x})$ for all $j \neq I$

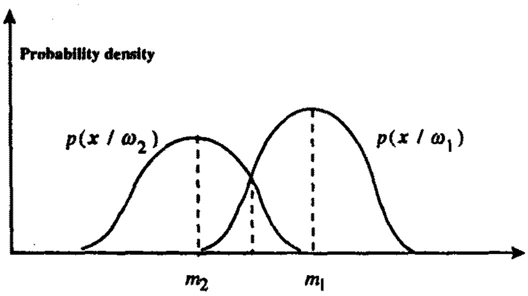
Bayes classifier for Gaussian pattern classes:

- Example : 1-D and 2 Classes problem

Bayes decision function is given by

$$d_i(x) = p(x/\omega_j)P(\omega_j)$$

$$= \frac{1}{\sqrt{2\pi}\sigma_j}\exp\left[-\frac{(x-m_j)^2}{2\sigma_j^2}\right]P(\omega_j) \qquad j=1,2$$



- Example : n-dimensional case

$$p(x/\omega_j) = \frac{1}{(2\pi)^{n/2}|C_j|^{1/2}}\exp\left[-\frac{1}{2}(x-m_j)^T C_j^{-1}(x-m_j)\right]$$

where the mean and the covariance matrix are given by

$$m_j = E_j[x] \quad \text{or} \quad \frac{1}{N_j}\sum_{x\in\omega_j} x$$

$$C_j = E_j\left[(x-m_j)(x-m_j)^T\right] \quad \text{or} \quad \frac{1}{N_j}\sum_{x\in\omega_j} xx^T - m_j m_j^T$$

IV. CLUSTERING METHODS

- Try to find natural groupings(clusters) of patterns
- Minimize a performance index
- Iterative processing
- Unsupervised learning(No labeled data are necessary)

1. K-means Algorithm

Performance index(sum of the squared errors) given by

$$J = \sum_{j=1}^{N_C} \sum_{x \in \omega_j} \left\| x - m_j \right\|^2$$

where N_C is the number of clusters, ω_j is the set of patterns belonging to the jth cluster, and

$$m_j = \frac{1}{N_j} \sum_{x \in \omega_j} x$$ is the sample mean vector of set ω_j

K-Means Algorithm

Specify the number of cluster N_C
Choose an initial set of cluster centers $m_j(0)$, $j=1, 2, ..., N_C$
Set an iteration count $I = 1$

Do Until $\|m_j(I+1) - m_j(I)\| < \varepsilon$ j=1, 2, ..., NC
  $I = I+1$
  For all patterns x
    Assign x to ω_j if $\|x-m_j\| < \| x-m_i \|$, $i=1,2, ..., N_C$ and $i \neq j$ where $\|*\|$ is any distance measure
  $m_j(I) = \frac{1}{N_j} \sum_{x \in \omega_j} x \quad j = 1,2,\ldots N_C$
End Until

2. Fuzzy K-means Algorithm

Performance index is given by

$$J_m = \sum_{k=1}^{Nn} \sum_{i=1}^{Nc} (\mu_{ik})^m (d_{ik})^2$$

where Nn is total number of patterns, Nc is the number of clusters, $m \in [1,\infty]$, $(d_{ik})^2$ is any distance measure, and μ_{ik} is the membership of the kth pattern belonging to the ith cluster.

Constraint:

$$\sum_{i=1}^{Nc} \mu_{ik} = 1 \text{ and } \mu_{ik} \in [0,1]$$

Fuzzy K-Means Algorithm

Specify the number of cluster N_C, and Fix m
Choose an initial set of cluster centers $\boldsymbol{m}_j(0)$, $j=1, 2, \ldots, N_C$
Set an iteration count $I = 1$

Do Until $\|\boldsymbol{m}_j(I+1) - \boldsymbol{m}_j(I)\| < \varepsilon$ j=1, 2, ..., NC
$I = I+1$
For all patterns $\boldsymbol{x}_k$

$$\mu_{ik} = \frac{1}{\sum_{j=1}^{Nc} \left(\frac{d_{ik}}{d_{jk}} \right)^{(2/m-1)}} \quad i = 1,2, \ldots, N_C$$

Update the cluter centers by

$$m_i(I) = \sum_{k=1}^{Nn} (\mu_{ik})^m \boldsymbol{x}_k \,/\, \sum_{k=1}^{Nn} (\mu_{ik})^m$$

End Until

Example: Trajectory of cluster centers in 3 classes problem.

Iteration count = 1, obj. fcn = 13.645593

Iteration count = 2, obj. fcn = 13.456014

Iteration count = 3, obj. fcn = 12.921680

Iteration count = 4, obj. fcn = 11.836673

Iteration count = 5, obj. fcn = 9.647694

Iteration count = 6, obj. fcn = 5.184969

Iteration count = 7, obj. fcn = 4.240686

Iteration count = 8, obj. fcn = 4.216349

Iteration count = 9, obj. fcn = 4.215861

Iteration count = 10, obj. fcn = 4.215851

Iteration count = 11, obj. fcn = 4.215851