

# XỬ LÝ ẢNH SỐ

## Chương 8

# Nhận dạng ảnh

- ❑ Nhận dạng ảnh
- ❑ Học có giám sát và không giám sát
- ❑ Một số mô hình nhận dạng ảnh
  - Mạng Hopfield
  - Mạng Kohonen

# Quá trình nhận dạng

- ❑ Quá trình nhận dạng dựa vào những mẫu học biết trước: **Học có giám sát**; ngược lại: **Học không giám sát**
- ❑ Quá trình nhận dạng
  - ❑ Lựa chọn mô hình biểu diễn đối tượng
  - ❑ Lựa chọn luật ra quyết định (phương pháp nhận dạng) và suy diễn quá trình học
  - ❑ Học nhận dạng
- ❑ Khi mô hình biểu diễn đã được xác định
  - ❑ Mô hình tham số
  - ❑ Mô hình cấu trúc
- ❑ Đến quá trình học nhằm cải thiện, điều chỉnh việc phân lớp
- ❑ Nhận dạng là tìm ra quy luật và thuật toán để gán đối tượng vào lớp của nó

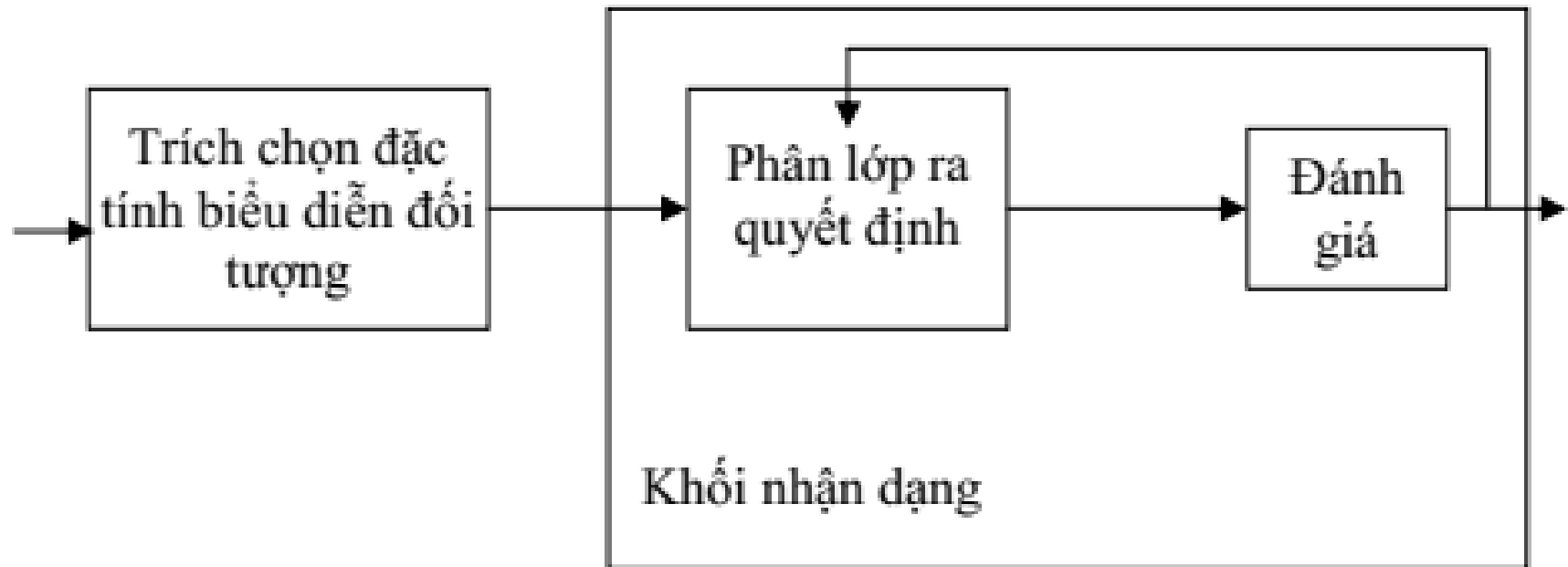
- ❑ Học có giám sát – supervised learning: dựa vào kiến thức biết trước.
- ❑ Sử dụng một thư viện mẫu chuẩn.
- ❑ Mẫu cần nhận dạng sẽ được so sánh với mẫu chuẩn để xác định mẫu đang đánh giá thuộc loại nào.
- ❑ Vấn đề chủ yếu là thiết kế hệ thống có thể so sánh đối tượng với mẫu nên tương đối rõ ràng về kết quả và dễ đánh giá hiệu quả

# Học có không giám sát

## unsupervised learning

- ❑ Học không giám sát phải tự định ra các lớp khác nhau và xác định đặc trưng của từng lớp
- ❑ Khó khăn để xác định tính chính xác: **số lớp**, những **đặc trưng của các lớp** đối tượng trong kiểu học này vì không có thư viện mẫu để so sánh
- ❑ Học không giám sát được phát triển nhằm hợp các nhóm theo một cách tối ưu.

# Cấu trúc 1 hệ thống nhận dạng



# Nhận dạng dựa theo không gian

- ❑ Trong kỹ thuật này thì đối tượng được biểu diễn bằng một vector nhiều chiều.
- ❑ Mỗi chiều là một tham số thể hiện một đặc điểm của đối tượng đó
- ❑ Một số khái niệm
  - ❑ Phân hoạch không gian
  - ❑ Hàm phân lớp hoặc hàm ra quyết định

# Phân hoạch không gian

- Giả sử không gian đối tượng  $X$  được biểu diễn bởi  $m$  vector  $X_i$ :

$$X = \{X_i, i = 1, 2, \dots, m\}$$

- $P$  được gọi là 1 phân hoạch không gian  $X$  thành các lớp  $C_i, C_i \subset X$  sao cho

$$C_i \cap C_j = \emptyset \quad i \neq j, \bigcup_{i=1}^m C_i = X$$

- Trong trường hợp thông thường thì không gian chỉ có thể được phân tách từng phần



- ❑ Để phân đối tượng ra các lớp ta cần xác định **số lớp** và **ranh giới giữa các lớp** đó
- ❑ Hàm phân lớp sẽ giúp phân biệt các lớp
- ❑ Nếu có  $k$  lớp thì phải có  $k-1$  hàm phân lớp

- Gọi  $\{g_i\}$  là lớp các hàm phân lớp. Lớp hàm này được định nghĩa như sau:

Nếu  $\forall i \neq k, g_k(X) > g_i(X)$  thì ta quyết định:  
 $\Rightarrow X \in \text{lớp } k.$

- Như vậy để phân biệt  $k$  lớp, cần  $k-1$  hàm phân biệt.
- Hàm phân lớp  $g$  thường là hàm tuyến tính:

$$g(X) = W_0 + W_1 X_1 + \dots + W_k X_k$$

Trong đó:

$W_i$  là các trọng số gán cho các thành phần  $X_i$ .

$W_0$  là trọng số hiệu chỉnh.

- ❑ Nếu  $g$  là tuyến tính, ta nói việc phân lớp là tuyến tính hay siêu phẳng (Hyperplane).
- ❑ Các hàm phân biệt thường được xây dựng dựa trên khái niệm khoảng cách hay dựa vào xác suất có điều kiện.
- ❑ Theo khoảng cách: Nếu khoảng cách giữa 2 đối tượng nhỏ hơn một ngưỡng  $\tau$  nào đấy thì coi 2 đối tượng là giống nhau và gộp chúng vào một lớp. Ngược lại, nếu khoảng cách lớn hơn ngưỡng, có nghĩa là chúng khác nhau và tách thành 2 lớp.

# Thuật toán nhận dạng không gian

(Tự học – Unsupervised learning)

- ❑ Một số thuật toán tiêu biểu
  - ❑ Thuật toán k trung bình (K-mean)
  - ❑ Thuật toán dựa vào khoảng cách lớn nhất
  - ❑ Thuật toán ISODATA
  - ❑ Thuật toán k láng giềng (K-nearest neighbour)

# Thuật toán nhận dạng không gian

(Tự học – Unsupervised learning)

## □ Thuật toán k trung bình (k - means)

- Đây là thuật toán dựa vào k phần tử đầu tiên trong không gian (xác định k lớp với k đã cho trước)
- Cách chọn k là tìm k lớp sao cho khoảng trung bình giữa các phần tử và tâm của lớp các phần tử này thuộc là nhỏ nhất
- Công thức:

$$Z_k = \frac{1}{N} \sum_{j=1}^N X_j, \quad X_j \in C_k$$

# Thuật toán nhận dạng không gian

(Tự học – Unsupervised learning)

## □ Thuật toán k trung bình: các bước

1. Chọn ngẫu nhiên K tâm (centroid) cho K cụm (cluster). Mỗi cụm được đại diện bằng các tâm của cụm.
2. Tính khoảng cách giữa các đối tượng đến K tâm (thường dùng khoảng cách Euclid).
3. Nhóm các đối tượng vào nhóm gần nhất.
4. Xác định lại tâm mới cho các nhóm.
5. Thực hiện lại bước 2 cho đến khi không có sự thay đổi nhóm nào của các đối tượng.

# Thuật toán nhận dạng không gian

(Tự học – Unsupervised learning)

□ **Thuật toán Khoảng cách lớn nhất:** Với một tập gồm  $m$  đối tượng:

- Khoảng cách giữa các đối tượng có thể được xác định như là đại lượng để phân lớp
- Khoảng cách lớn nhất ứng với phần tử xa nhất tạo nên lớp mới
- Sự phân lớp sẽ hình thành dần dần dựa vào việc tính khoảng cách giữa các đối tượng và các lớp

# Thuật toán nhận dạng không gian

(Tự học – Unsupervised learning)

## □ Thuật toán Khoảng cách lớn nhất: các bước

### Bước 1:

Chọn hạt nhân ban đầu:  $X_i \in C_1$  là lớp ứng với hàm  $g_1$  với  $Z_1$  là phần tử trung tâm (phần tử có khoảng cách trung bình tới các phần tử khác trong  $g_1$ )

Tính tất cả các khoảng cách  $D_{j1} = D(X_j, Z_1)$  với  $j=1..m$

Tìm  $D_{k1}$  lớn nhất.  $X_k$  là phần tử xa nhất của nhóm  $g_1$ ,  $X_k$  trở thành phần tử trung tâm  $Z_2$  của lớp mới  $g_2$ .

Tính  $d_1 = D_{12} = D(Z_1, Z_2)$



# Thuật toán nhận dạng không gian

(Tự học – Unsupervised learning)

□ Thuật toán Khoảng cách lớn nhất: các bước

Bước 2:

Tính các khoảng cách  $D_{j1}$ ,  $D_{j2}$  với  $j=1..m$

Đặt  $D_k$  là khoảng cách lớn nhất

Điều kiện kết thúc

Nếu  $D_k < \theta d_1$  kết thúc thuật toán (phân lớp xong) Nếu không sẽ tạo nên lớp  $g_3$  có  $X_k$  là phần tử trung tâm  $Z_3$

Tính  $d_3 = (D_{12} + D_{13} + D_{23})/3$

Thuật toán lặp cho đến khi thỏa mãn đk kết thúc

# Thuật toán nhận dạng không gian

(Tự học – Unsupervised learning)

## □ Thuật toán Isodata:

- ISODATA: Iterative Self Organizing Data Analysis
- Thuật toán này tương đối mềm dẻo và không cố định các lớp

# Thuật toán nhận dạng không gian

(Tự học – Unsupervised learning)

## □ Thuật toán Isodata: các bước

- Lựa chọn phân hoạch dựa vào các tâm bất kỳ (kết quả không phụ thuộc vào tâm ban đầu này)
- Phân vùng bằng cách đặt các điểm vào lớp có tâm gần nhất dựa vào khoảng cách
- Tách đôi lớp ban đầu nếu khoảng cách lớn hơn ngưỡng  $t_1$
- Xác định phân hoạch mới dựa vào các tâm vừa xác định cho đến khi không có tâm mới
- Nhóm các vùng theo ngưỡng  $t_2$

Lặp các bước trên cho đến khi thỏa mãn tiêu chuẩn phân hoạch.

# Thuật toán nhận dạng không gian

(Tự học – Unsupervised learning)

## □ Thuật toán k láng giềng:

- Thuật toán này dựa vào vị trí của đối tượng cần xét để xác định lớp mà đối tượng đó thuộc về là lớp nào
- Các mẫu sẽ được biểu diễn trên không gian đối tượng
- Mỗi đối tượng cần xác định lớp sẽ tìm k mẫu gần nhất xung quanh
- Đối tượng đầu vào này sẽ thuộc lớp có số lượng lớn nhất trong số k láng giềng

# Thuật toán nhận dạng không gian

(Tự học – Unsupervised learning)

## □ Thuật toán k láng giềng:

- Vì thuật toán chọn số đông để quyết định việc phân lớp nên thông thường lớp lớn nhất trong tập mẫu thường có xu hướng thống trị
- Để làm giảm việc này thì trọng số liên quan đến khoảng cách có thể được thêm vào, và mẫu càng xa sẽ có ảnh hưởng ít hơn so với mẫu ở gần

# Thuật toán nhận dạng theo cấu trúc

- ❑ Ngoài cách biểu diễn định lượng thì còn tồn tại kiểu đối tượng định tính
- ❑ Ví dụ mối quan hệ giữa các đối tượng hoặc dạng của đối tượng
- ❑ Giả sử đối tượng có thể được biểu diễn bởi một chuỗi ký tự và mỗi ký tự sẽ thể hiện một đặc tính
- ❑ Hàm phân biệt sẽ là hàm logic nhận diện các từ có cùng độ dài

# Thuật toán nhận dạng theo cấu trúc

- ❑ Có hai giai đoạn trong quy trình nhận diện:
  - Xác định quy tắc xây dựng
  - Xác định các dạng dựa vào các quy tắc đó
- ❑ Trong hai giai đoạn thì xác định quy tắc xây dựng là rất khó khăn và là vấn đề chính cần giải quyết trong quy trình nhận diện dạng này

# Nhận diện dựa trên mạng Nơron

- ❑ Mạng nơron bao gồm nhiều phần tử xử lý đơn giản hoạt động song song.
- ❑ Tính năng của hệ thống phụ thuộc vào cấu trúc liên kết giữa các nơron và trọng số của các liên kết này
- ❑ Mạng nơron có thể học từ dữ liệu mẫu và tổng quát hóa dựa trên các dữ liệu mẫu học. Bản chất quá trình học là việc hiệu chỉnh các tham số của các nơron để đầu mạng dự đoán đúng dữ liệu học.
- ❑ Trong mạng nơron, các nơron đón nhận tín hiệu vào gọi là nơron vào và các nơron đưa thông tin ra gọi là nơron ra.

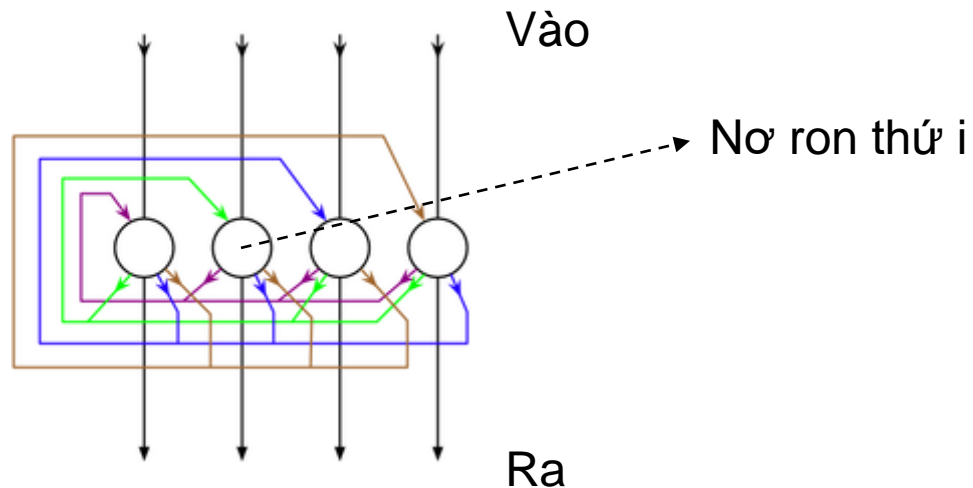


## □ Mạng Hopfield

- Mạng Hopfield là mạng nơ-ron 1 lớp
- Ánh xạ dữ liệu tín hiệu vào sang tín hiệu ra theo kiểu tự kết hợp
- Nếu tín hiệu vào là  $X$  thuộc miền  $D$  thì tín hiệu ra  $Y$  cũng thuộc miền  $D$
- Mạng Hopfield mô phỏng khả năng hồi tưởng của não người (nhận ra người quen khi nhận ra các nét quen trên khuôn mặt)

## □ Mạng Hopfield

- Mạng Hopfield có một lớp ra có kích thước bằng kích thước tín hiệu vào, liên kết nơ-ron là đầy đủ.
- Mạng Hopfield yêu cầu tín hiệu vào có giá trị lưỡng cực -1 và 1



## □ Mạng Hopfield

- Hàm kích hoạt tại các nơ ron là hàm dấu:

$$out_j = sign(Net_j) = sign\left(\sum_{i=1}^m w_{ji} x_i\right)$$

$w_{ij}$  là trọng số của liên kết giữa nơ ron thứ  $i$  với nơ ron thứ  $j$

## □ Mạng Hopfield

- Mạng Hopfield là mạng học có giám sát
- Giả sử có  $p$  mẫu học tương ứng với các vector tín hiệu vào  $X_s$ ,  $s = 1..p$
- Mạng cần xác định bộ trọng số  $W$  sao cho

$$X_s = f(X_s, W) \text{ với mọi } s = 1..p$$

- Ta xây dựng ma trận trọng số  $W$  như sau:

$$w_{ij} = \begin{cases} \sum_{s=1}^p x_{sj} x_{si}, i \neq j \\ 0, i = j \end{cases}$$

## □ Mạng Hopfield: Huấn luyện mạng

- Giả sử đưa vào mạng vector tín hiệu  $X$
- Việc tính toán đầu ra  $Y$  cho tín hiệu  $X$  là quá trình lặp lại:
  - Đặt  $X_0 = X$
  - Tính  $Y_t$  là tín hiệu đầu ra tương ứng với  $X_t$  lan truyền trong mạng một lần
  - Nếu  $Y_t$  khác  $X_t$  thì tiếp tục với  $t = t+1$  và  $X_{t+1} = Y_t$
  - Nếu  $Y_t = X_t$  thì dừng lại và kết quả của mạng là  $Y_t$
  - $W$  không thay đổi trong quá trình tính  $Y$

- ❑ **Mạng Hopfield:** Một số tình huống nảy sinh trong huấn luyện mạng
  - Mạng không dừng:
    - Mạng có thể đưa ra luân phiên một vài mẫu học hoặc ảnh ngược của chúng
  - Mạng dừng và  $X_t = X$ 
    - $X$  đã được đoán đúng dựa trên mẫu học,  $X$  có thể là một trong các mẫu đã học

- ❑ **Mạng Hopfield:** Một số tình huống nảy sinh trong huấn luyện mạng
  - Mạng dừng với  $X_t$  là  $X_s$  nào đó trong mẫu đã học
  - Mạng đã phục hồi nguyên dạng của  $X$  chính là  $X_s$
  - Mạng dừng với  $X_t$  không thuộc mẫu đã học
  - Chỉ ra một vector mới có thể dùng để cập nhật trọng số
  - Mạng dừng với  $X_t$  là ảnh ngược ở trên.

## □ Mạng Kohonen

Bản đồ Kohonen hoặc bản đồ tự tổ chức

- Mạng Kohonen tập trung vào mối liên hệ có tính cấu trúc trong các vùng lân cận hoặc trong toàn thể không gian mẫu
- Trong mạng Kohonen các vector tín hiệu vào gần nhau sẽ được ánh xạ sang các nơ ron trong mạng lân cận nhau



## □ Mạng Kohonen

### Bản đồ Kohonen hoặc bản đồ tự tổ chức

- Mạng Kohonen rất gần với mạng sinh học về cấu tạo lẫn cơ chế học
- Mạng Kohonen có một lớp kích hoạt là các nơ-ron được phân bố trong mặt phẳng hai chiều kiểu lưới vuông hoặc lục giác
- Phân bố này làm cho mỗi nơ-ron có cùng số nơ-ron trong từng lớp láng giềng và các đầu vào tương tự nhau sẽ kích hoạt các nơ-ron gần nhau (không gian 2 chiều sẽ phải được cuộn để thành một không gian liên tiếp)

## □ Mạng Kohonen

Bản đồ Kohonen hoặc bản đồ tự tổ chức



Lưới các nơ-ron trong mặt phẳng 2 chiều

## □ Mạng Kohonen

Bản đồ Kohonen hoặc bản đồ tự tổ chức

- Tất cả các nơ ron ở lớp kích hoạt được nối đầy đủ với lớp vào và mỗi kết nối này sẽ có một trọng số
- Các nơ ron trên lớp kích hoạt chỉ nối với các lớp lân cận nên khi có tín hiệu đầu vào sẽ chỉ tạo ra kích hoạt địa phương
- Vùng được kích hoạt sẽ chỉ ra vị trí của tín hiệu đầu vào trong không gian đối tượng

## □ Mạng Kohonen

### Bản đồ Kohonen hoặc bản đồ tự tổ chức

- Việc học của mạng kohonen dựa trên kỹ thuật cạnh tranh không cần mẫu học.
- Trọng số của các nơ-ron được khởi tạo bằng một số nhỏ, bất kỳ
- Mạng được học qua một lượng lớn mẫu gần nhau và thường nhiều lần theo các vòng lặp
- Tại một thời điểm chỉ có một nơ-ron duy nhất  $C$  trong lớp kích hoạt được lựa chọn với nguyên tắc là nơ-ron  $C$  có vector trọng số gần với tín hiệu vào nhất

## □ Mạng Kohonen

Bản đồ Kohonen hoặc bản đồ tự tổ chức

- Sau khi có nơron  $C$ , các trọng số  $w_{ci}$  được hiệu chỉnh nhằm cho gần với trọng số mong muốn (đầu vào)
- Nếu tín hiệu vào  $x_{si}$  và trọng số  $w_{ci}$  tạo kết quả ra quá lớn thì phải giảm trọng số và ngược lại

## □ Mạng Kohonen

### Bản đồ Kohonen hoặc bản đồ tự tổ chức

- Tỷ lệ giảm hoặc tăng trọng số sẽ nhỏ dần theo thời gian và khoảng cách sai số giữa vector đầu vào và vector trọng số với công thức:

$$w_v(s+1) = w_v(s) + \theta(u, v, s) \alpha(s) (D(t) - w_v(s))$$

- Với:
  - $s$  là thứ tự bước
  - $D(t)$  là vector đầu vào  $t$
  - $u$  là số thứ tự của nơ ron  $C$  được chọn
  - $\theta(u, v, s)$  là hàm tỷ lệ khoảng cách giữa nơ ron  $v$  và  $u$  (có thể là hàm ngưỡng hoặc hàm Gaussian)

## □ Mạng Kohonen

### Bản đồ Kohonen hoặc bản đồ tự tổ chức

- Mạng Kohonen được huấn luyện dựa vào sự kích hoạt địa phương và các nơ-ron lân cận nên các đối tượng gần nhau sẽ kích hoạt các nơ-ron gần nhau
- Khi chuyển một tín hiệu đầu vào vào mạng thì một vùng nơ-ron sẽ được kích hoạt cho chúng ta thấy tín hiệu đầu vào này thuộc vùng dữ liệu nào

- ❑ Nhận dạng ảnh
- ❑ Học có giám sát và không giám sát
- ❑ Một số mô hình nhận dạng ảnh
  - Mạng Hopfield
  - Mạng Kohonen