BỘ GIÁO DỤC VÀ ĐÀO TẠO BỘ QUỐC PHÒNG

HỌC VIỆN KỸ THUẬT QUÂN SỰ

NGUYỄN TRUNG DŨNG

NGHIÊN CỨU THUẬT TOÁN ĐỒNG PHÂN CỤM MỜ CHO BÀI TOÁN PHÂN ĐOẠN ẢNH

Chuyên ngành: Khoa Học Máy Tính

LUẬN VĂN THẠC SĨ KỸ THUẬT

Hà Nội - Năm 2014

BỘ GIÁO DỤC VÀ ĐÀO TẠO BỘ QUỐC PHÒNG

HỌC VIỆN KỸ THUẬT QUÂN SỰ

NGUYỄN TRUNG DŨNG

NGHIÊN CỨU THUẬT TOÁN ĐỒNG PHÂN CỤM MỜ CHO BÀI TOÁN PHÂN ĐOẠN ẢNH

Chuyên ngành: Khoa Học Máy Tính

Mã số: .....

LUẬN VĂN THẠC SĨ KỸ THUẬT

Hà Nội - Năm 2014

CÔNG TRÌNH ĐƯỢC HOÀN THÀNH TẠI

HỌC VIỆN KỸ THUẬT QUÂN SỰ

Cán bộ hướng dẫn chính: PGS. TS Ngô Thành Long

Cán bộ chấm phản biện 1:.....

Cán bộ chấm phản biện 2:.....

Luận văn thạc sĩ được bảo vệ tại:

HỘI ĐỒNG CHẤM LUẬN VĂN THẠC SĨ

HỌC VIỆN KỸ THUẬT QUÂN SỰ

Ngày ... tháng ... năm 2014

HỌC VIỆN KỸ THUẬT QUÂN SỰ
PHÒNG SAU ĐẠI HỌC

CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM
Độc lập – Tự do – Hạnh phúc

Hà Nội, ngày tháng năm 2014

NHIỆM VỤ LUẬN VĂN THẠC SĨ

Họ tên học viên: Nguyễn Trung Dũng Giới tính: Nam

Ngày, tháng, năm sinh: 15/10/1989 Nơi sinh: Hà Nội

Chuyên ngành: Khoa Học Máy Tính Mã số: ……

I- TÊN ĐỀ TÀI: "NGHIÊN CỨU THUẬT TOÁN ĐỒNG PHÂN CỤM MỜ CHO BÀI TOÁN PHÂN ĐOẠN ẢNH"

II- NHIỆM VỤ VÀ NỘI DUNG:

Ứng dụng thuật toán đồng phân cụm mờ vào phân đoạn ảnh.

+ Phân cụm ảnh bằng thuật toán đồng phân cụm mờ (FCCI)

+ Hiển thị, lưu trữ các cụm của ảnh và ảnh sau khi được phân đoạn.

III- NGÀY GIAO NHIỆM VỤ:

IV- NGÀY HOÀN THÀNH NHIỆM VỤ:

V- CÁN BỘ HƯỚNG DẪN: PGS. TS Ngô Thành Long

CÁN BỘ HƯỚNG DẪN **CHỦ NHIỆM BỘ MÔN QL CHUYÊN NGÀNH**

Nội dung và đề cương luận văn thạc sĩ đã được Hội đồng chuyên ngành thông qua.

Ngày tháng năm 2014

TRƯỞNG PHÒNG SĐH **TRƯỞNG KHOA QL NGÀNH**

MỤC LỤC

Trang

Tóm tắt luận văn:

Họ và tên học viên: **Nguyễn Trung Dũng**

Chuyên ngành: Khoa Học Máy Tính Khóa: 25

Cán bộ hướng dẫn: PGS.TS Ngô Thành Long

Tên đề tài: Nghiên cứu thuật toán đồng phân cụm mờ và áp dụng vào cho bài toán phân đoạn ảnh.

Tóm tắt: Luận văn giới thiệu thuật toán đồng phân cụm mờ(FCCI) nhằm thực hiện phân cụm bằng cách sử dụng đồng thời 2 hàm đối tượng và đặc trưng.Đồng thời tiến hành tính toán số tâm cụm bằng giải thuật Xie-Benni.Các thử nghiệm cho thấy thuật toán FCCI thực hiện phân đoạn tốt trên dữ liệu nhiều chiều,cụ thể ở đây là ảnh đa màu sắc.

.

DANH MỤC CÁC KÍ HIỆU

CNTT	Công nghệ thông tin
CSDL	Cơ sở dữ liệu
εFCM	Phân cụm mờ nhạy cảm với nhiễu
FCM	Phân cụm mờ loại một
FL	Logic mờ
FOU	Footprint of Uncertainty
FS	Tập mờ
GC	Trọng tâm tổng quát
IT2FCM	Phân cụm mờ loại hai khoảng
IT2FS	Tập mờ loại hai khoảng
LMF	Hàm thuộc dưới
MF	Hàm thuộc
PCDL	Phân cụm dữ liệu
T2FCM	Phân cụm mờ loại hai
T2FS	Tập mờ loại hai
UMF	Hàm thuộc trên
FCCI	Đồng phân cụm

DANH MỤC HÌNH VẼ

MỞ ĐẦU

Logic mờ được công bố lần đầu tiên tại Mỹ vào năm 1965 bởi giáo sư L.Zadeh. Kể từ đó, Logic mờ đã có bước phát triển mạnh mẽ trong nhiều lĩnh vực và các ứng dụng thực tế khác nhau. Đặc biệt, việc ứng dụng Logic mờ trong lĩnh vực xử lý ảnh đã đem lại những hiệu quả rõ rệt. Bởi vì, với việc áp dụng Logic mờ vào trong xử lý ảnh, ta đã phần nào xử lý được những yếu tố không chắc chắn thường xuyên xảy ra trong xử lý ảnh, bởi vì đầu vào ảnh thường có nhiễu và các đối tượng trong ảnh thường không rõ ràng và nằm chồng lên nhau. Chính vì vậy, việc ứng dụng Logic mờ vào xử lý ảnh đã trở thành hướng nghiên cứu và quan tâm của rất nhiều nhà khoa học cũng như người sử dụng.

Cùng với sự phát triển ngày càng mạnh mẽ của khoa học kĩ thuật trong một vài thập kỷ gần đây, xử lý ảnh tuy là một ngành khoa học còn tương đối mới mẻ so với nhiều ngành khoa học khác nhưng hiện nay nó đang là một trong những lĩnh vực phát triển rất nhanh và thu hút sự quan tâm đặc biệt từ các nhà khoa học, thúc đẩy các trung tâm nghiên cứu, ứng dụng về lĩnh vực hấp dẫn này.Để xử lý được một bức ảnh thì phải trải qua nhiều khâu khác nhau tùy theo mục đích của việc xử lý, nhưng khâu quan trọng và khó khăn nhất đó là phân đoạn ảnh. Trong một số lượng lớn các ứng dụng về xử lý ảnh và hiển thị máy tính, phân đoạn đóng vai trò chính yếu như là bước đầu tiên trước khi áp dụng các thao tác xử lý ảnh mức cao hơn như: nhận dạng, giải thích ngữ nghĩa, và biểu diễn ảnh.

Phân đoạn ảnh là một thao tác ở mức thấp trong toàn bộ quá trình xử lý ảnh. Quá trình này thực hiện việc phân vùng ảnh thành các vùng rời rạc và đồng nhất với nhau hay nói cách khác là xác định các biên của các vùng ảnh đó. Các vùng ảnh đồng nhất này thông thường sẽ tương ứng với toàn bộ hay

từng phần của các đối tượng thật sự bên trong ảnh. Vì thế, trong hầu hết các ứng dụng của lĩnh vực xử lý ảnh, phân đoạn ảnh luôn đóng một vai trò cơ bản và thường là bước tiền xử lý đầu tiên trong toàn bộ quá trình trước khi thực hiện các thao tác khác ở mức cao hơn như nhận dạng đối tượng, biểu diễn đối tượng, nén ảnh dựa trên đối tượng, hay truy vấn ảnh dựa vào nội dung … Trước đây, các phương pháp phân vùng ảnh được đưa ra chủ yếu làm việc trên các ảnh mức xám do các hạn chế về phương tiện thu thập và lưu trữ. Ngày nay, cùng với sự phát triển về các phương tiện thu nhận và biểu diễn ảnh, các ảnh màu đã hầu như thay thế hoàn toàn các ảnh mức xám trong việc biểu diễn và lưu trữ thông tin do các ưu thế vượt trội hơn hẳn so với ảnh mức xám. Do đó, các kỹ thuật, thuật giải mới thực hiện việc phân vùng ảnh trên các loại ảnh màu liên tục được phát triển để đáp ứng các nhu cầu mới.

Với đề tài *"Nghiên cứu thuật toán đồng phân cụm mờ cho bài toán phân đoạn ảnh"*, luận văn sẽ trình bày một số vấn đề về phân cụm dữ liệu và việc ứng dụng Logic mờ vào phân cụm dữ liệu. Trong đó, luận văn tập trung vào việc sử dụng thuật toán đồng phân cụm mờ(FCCI) để thực hiện phân đoạn ảnh.

Phân cụm là một công cụ toán học dùng để phát hiện cấu trúc hoặc các mẫu nào đó trong tập dữ liệu, theo đó có đối tượng bên trong cụm dữ liệu thể hiện bậc tương đồng nhất định.Nói cách khác, phân cụm dữ liệu là quá trình nhóm một tập các đối tượng tương tự nhau trong tập dữ liệu vào các cụm sao cho các đối tượng thuộc cùng một cụm là tương đồng còn các đối tượng thuộc các cụm khác nhau sẽ không tương đồng. Ngoài ra phân cụm dữ liệu còn có thể được sử dụng như một bước tiền xử lí cho các thuật toán khai phá dữ liệu khác như là phân loại và mô tả đặc điểm, có tác dụng trong việc phát hiện ra các cụm.Kỹ thuật phân cụm được áp dụng trong rất nhiều lĩnh vực như khai phá dữ liệu, nhận dạng mẫu, xử lý ảnh.

Đây là hướng nghiên cứu có triển vọng vì phân đoạn ảnh là một ứng dụng đóng vai trò cơ sở, nền tảng để việc thực hiện các ứng dụng xử lý ảnh như nhận dạng, giải mã

Luận văn được trình bày trong 3 chương:

Chương 1: Giới thiệu tổng quan về Logic mờ

Chương 2: Giới thiệu các phương pháp phân cụm dữ liệu.

Chương 3: Trình bày thuật toán đồng phân cụm mờ và kết quả

Kết luận : Tóm tắt các vấn đề được tìm hiểu trong luận văn và các vấn đề liên quan trong luận văn, đưa ra phương hướng nghiên cứu tiếp theo.

Chương 1
TỔNG QUAN VỀ LOGIC MỜ

1.1. Tập mờ loại một

Logic mờ (FL) theo nghĩa rộng mà ngày nay được dùng rộng rãi, có cùng nghĩa với lý thuyết tập mờ.

Trong mục này sẽ trình bày những khái niệm cơ bản liên quan tới tập mờ như: khái niệm tập mờ, hàm thuộc (membership funtions)…

Một tập cổ điển là một tập có biên xác định.

Ví dụ 1.1: Tập cổ điển A là tập của các số thực lớn hơn 1.8 và có thể miêu tả như sau: A = {x | x > 1.8}

Với cách miêu tả trên thì rất rõ ràng, tập A có biên là 1.8, nếu x lớn hơn 1.8 thì nằm trong tập A, ngược lại thì x không nằm trong tập A. Tuy vậy các tập cổ điển không phản ánh được các khái niệm và suy nghĩ của con người. Ví dụ: định nghĩa lại tập A trong ví dụ 1.1, tập A là tập các người cao và x là chiều cao. Như vậy, với miêu tả biểu thức toán học ở ví dụ 1.1 thì những người có chiều cao lớn hơn 1.8 m mới được gọi là người cao. Điều này không hợp lý trong thực tế vì nếu người nào có chiều cao 1.7999 m thì không được gọi là người cao. Do vậy, để miêu tả tập người cao thì không thể dùng tập cổ điển.

Như vậy tập mờ là tập không có biên xác định, đây là một đặc điểm trái ngược với tập cổ điển.

1.1.1. Định nghĩa tập mờ loại một

Cho X là không gian của các đối tượng x, x là một đối tượng (phần tử) thuộc X. Một tập cổ điển A, $A \in X$, là tập gồm các phần tử $A \in X$, như vậy với mỗi $x \in X$ có thể thuộc tập A hoặc không thuộc tập A.

Hàm đặc tính (characteristic funtions) cho mỗi đối tượng $x \in X$ có quan hệ với tập A như sau: Tập cổ điển A là một tập của các cặp phần tử có bậc (x,0) với $x \notin A$ hoặc (x, 1) với $x \in A$. Với cách định nghĩa trên, có thể miêu tả tập cổ điển A thông qua hàm đặc tính:

$$A = \{ x, \mu_A(x) \mid x \in X \}$$

Trong đó: $\mu_A(x)$ là hàm đặc tính được xác định:

$$\mu_A(x) = \begin{cases} 0, x \notin A \\ 1, x \in A \end{cases} \text{ với } \forall x \in X$$

Định nghĩa 1.1: Tập mờ và hàm thuộc

Nếu X là một tập hợp các đối tượng x, x biểu diễn chung cho đối tượng, khi đó một tập mờ $A \subseteq X$ được định nghĩa như một tập của các cặp phần tử có bậc: $A = \{ x, \mu_A(x) \mid x \in X \}$

Ở đây: $\mu_A(x)$ được gọi là hàm thuộc (MF) cho tập mờ A. MF ánh xạ mỗi phần tử $x \in X$ tới độ thuộc giữa 0 và 1 của MF.

Với định nghĩa trên, không giống như tập cổ điển, tập mờ có hàm đặc tính (theo nghĩa của tập cổ điển) cho phép có giá trị nằm giữa 0 và 1. Như vậy định nghĩa của tập mờ là một mở rộng đơn giản của định nghĩa tập cổ điển trong đó hàm thuộc có độ thuộc giữa 0 và 1. Nếu giá trị của hàm thuộc $\mu_A(x)$ được đưa về chỉ có 0 và 1, khi đó A chính là tập cổ điển và $\mu_A(x)$ là một hàm đặc tính của A.

Thông thường X được xem như là tập nền. X có thể là các đối tượng rời rạc (có thứ tự hoặc không thứ tự) hoặc không gian liên tục.

Một tập mờ A có thể biểu diễn:

$\sum_{x_i \in X} \mu_A(x_i) / x_i$, nếu X là tập các đối tượng rời rạc

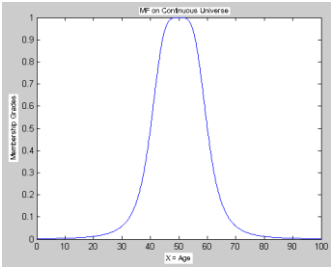
$\int_X \mu_A(x)/x$, nếu X là không gian liên tục (thường là trên R)

Ví dụ 1.2: Cho $X = R^+$ là tập tuổi cho người. Và tập mờ B = "Khoảng 50 tuổi", khi đó B được biểu diễn như sau:

$$B = \{x, \mu_B(x) \mid x \in X\}$$

Trong đó: $\mu_B(x)$ được xác định: $\mu_B(x) = \frac{1}{1+\left(\frac{x-50}{10}\right)^4}$ với $x \in X$

Và đồ thị của nó có dạng như hình 1.1 dưới đây:



Hình 1 Đồ thị MF

1.1.2. Biểu thức và tham số của một số hàm thuộc.

1.2.2.1. Hàm thuộc một chiều

Hàm thuộc một chiều là hàm chỉ có một đầu vào. Do vậy, các hàm đưa ra dưới đây sẽ được hiểu ngầm định là luôn luôn có một đầu vào.

Định nghĩa 1.2: Hàm thuộc Triangular

Một hàm thuộc triangular được đưa ra bởi 3 tham số {a, b, c} (với a < b < c) như sau:

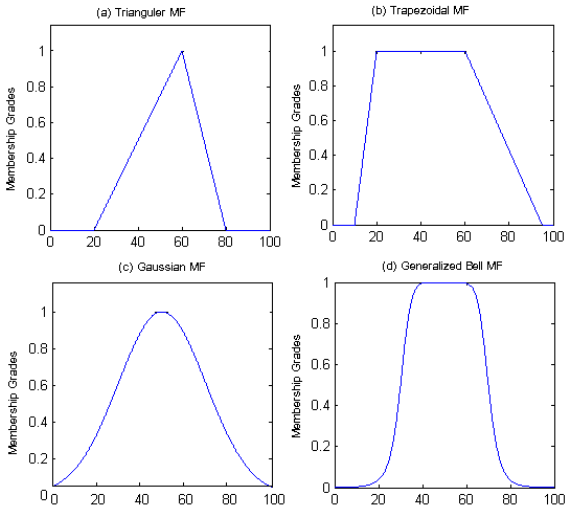
$$triangle(x,a,b,c) = \begin{cases} 0, & x \le a \\ \frac{x-a}{b-a}, & a \le x \le b \\ \frac{c-x}{c-b}, & b \le x \le c \\ 0, & c \le x \end{cases}$$

Bằng cách dùng min và max, người ta đã đưa biểu diễn biểu thức trên như sau:

$$triangle\ \ x,a,b,c\ \ = \max\left(\min\left(\frac{x-a}{b-a},\frac{c-x}{c-b},0\right)\right)$$

Ở đây: Các tham số {a, b, c} xác định tọa độ x của ba góc của hàm thuộc Triangular.

Hình 1.2(a) dưới đây minh họa hàm thuộc Triangular được định nghĩa bởi triangular(x; 20, 60, 80).



Hình 2 Các ví dụ của bốn loại hàm thuộc

Hình 2. Các ví dụ của bốn loại hàm thuộc: (a) Triangle(x; 20, 60, 80); (b) Trapezoid(x; 10, 20, 60, 95); (c) Gaussian(x; 50, 20); (d) bell(x; 20, 4, 50)

Định nghĩa 1.3: Hàm thuộc Trapezoidal (Hình thang)

Một hàm thuộc trapezoidal được đưa ra bởi 4 tham số {a, b, c, d} (với a < b < c < d) như sau:

$$trapezoid(x;a,b,c,d)=\begin{cases}0, & x\le a\\ \dfrac{x-a}{b-a}, & a\le x\le b\\ 1, & b\le x\le c\\ \dfrac{d-x}{d-c}, & c\le x\le d\\ 0, & d\le x\end{cases}$$

Bằng cách dùng min và max, người ta đã đưa ra biểu diễn biểu thức trên như sau:

$$trapezoid(x;a,b,c,d)=\max\left(\min\left(\frac{x-a}{b-a},\frac{d-x}{d-c}\right),0\right)$$

Ở đây: Các tham số {a, b, c, d} xác định tọa độ x của bốn góc của hàm thuộc Trapezoidal.

Hình 1.2(b) minh họa hàm thuộc Trapezoidal được định nghĩa bởi trapezoidal(x; 10, 20, 60, 95).

Định nghĩa 1.4: Hàm thuộc Gaussian

Hàm thuộc Gaussian được đưa ra bởi 2 tham số c,∂ :

$$gaussian\ x,c,\partial = e^{-\frac{1}{2}\left(\frac{x-c}{\partial}\right)^2}$$

Ở đây: c miêu tả vị trí trọng tâm và ∂ xác định độ rộng của hàm thuộc Gaussian.

Hình 1.2(c) minh họa hàm thuộc Gaussian được định nghĩa bởi gaussian (x; 50, 20).

Định nghĩa 1.5: Hàm thuộc bell – hình chuông

Hàm thuộc bell – hình chuông được đưa ra bởi 3 tham số {a, b, c}:

$$bell\ x;a,b,c = \frac{1}{1+\left|\frac{x-c}{a}\right|^{2b}}$$

Ở đây: b luôn luôn dương và tham số c định vị trí trọng tâm của đường cong.

Hình 1.2(d) minh họa hàm thuộc Bell được định nghĩa bởi bell(x; 20, 4, 50).

Định nghĩa 1.6: Hàm thuộc sigmoidal

Hàm thuộc sigmoidal được định nghĩa bởi:

$$sig(x;a,c)=\frac{1}{1+\exp\left[-a(x-c)\right]}$$

Hàm này phụ thuộc vào dấu của tham số a, có tính mở trái và phải. Do vậy, nó gần như miêu tả các khái niệm "$+\infty$" và "$-\infty$". Hàm này được khai thác rộng rãi. Tuy nhiên để khai thác được cần biết cách kết hợp các hàm sigmoidal lại với nhau. Ví dụ dưới đây đưa ra hai cách kết hợp các hàm sigmoidal để tạo ra các hàm thuộc có tính đóng và tính không đối xứng.

Định nghĩa 1.7: Hàm thuộc left - right

Hàm thuộc left – right được đưa ra bởi 3 tham số α,β,c :

$$LR\ x;c,\alpha,\beta\ =\begin{cases}F_L\left(\dfrac{c-x}{\alpha}\right), x\le c\\ F_R\left(\dfrac{x-c}{\beta}\right), x\ge c\end{cases}$$

Ở đây: $F_L\ x$ và $F_R\ x$ là các hàm giảm đơn điệu trên $0,\infty$ với $F_L\ 0\ =F_R\ 0\ =1$ và $\lim_{x\to\infty}F_L(x)=\lim_{x\to\infty}F_R(x)=0$.

1.2.2.2. Một số hàm thuộc hai chiều

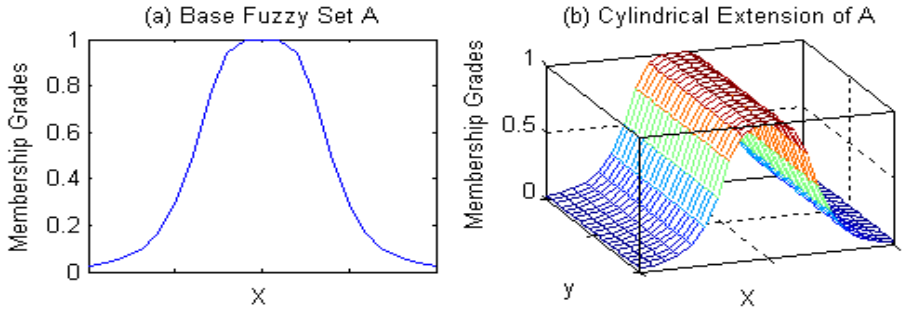
Hàm thuộc hai chiều là hàm có hai đầu vào. Cách cơ bản để mở rộng hàm thuộc một chiều thành hàm hai chiều là thông qua mở rộng trụ (cylindrical extension), được định nghĩa như sau:

Định nghĩa 1.8: Mở rộng trụ của hàm thuộc một chiều

Nếu A là tập mờ trong X, khi đó mở rộng trụ của A trong $X\times Y$ là tập mờ C(A) được định nghĩa:

$$C(A) = \int_{X \times Y} \mu_A(x)/(x, y)$$

Hình 1.3 dưới đây minh họa mở rộng trụ của tập mờ A.

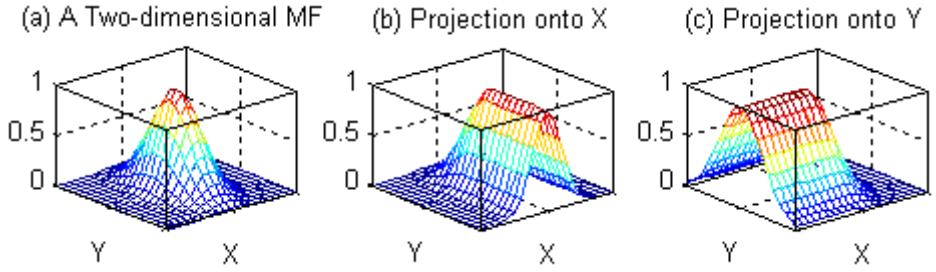


Hình 3 Tập mờ cơ sở A và Mở rộng trụ C(A) của A

Định nghĩa 1.9: Các phép chiếu của tập mờ

Cho R là tập mờ hai chiều trên $X \times Y$. Khi đó các phép chiếu trên X và Y được định nghĩa tương ứng:

$$R_X = \int_X \left[\max_y \mu_R(x, y) \right] / x$$



Hình 4 Tập mờ hai chiều R

Nói chung, hàm thuộc hai chiều được chia thành hai nhóm: kết hợp và không kết hợp. Nếu hàm thuộc hai chiều có thể được biểu diễn thông qua hai hàm thuộc một chiều thì khi đó nó thuộc nhóm kết hợp. Ngược lại thì là nhóm không kết hợp.

Ví dụ 1.5: Hàm thuộc hai chiều thuộc nhóm kết hợp và không kết hợp

Giả sử tập mờ A = "(x, y) is near (3, 4)" được định nghĩa bởi:

$$\mu_A(x,y) = \exp\left[-\left(\frac{x-3}{2}\right)^2 - \left(y-4\right)^2\right]$$

Đây là hàm thuộc hai chiều thuộc nhóm kết hợp. Do vậy nó có thể được phân tích thành hai hàm thuộc một chiều như sau:

$$\mu_A(x,y) = \exp\left[-\left(\frac{x-3}{2}\right)^2\right]\exp\left[-\left(\frac{y-4}{1}\right)^3\right]$$

$$= gaussian(x;3,2)gaussian(y;4,1)$$

Với cách tách như trên thì bây giờ ta có thể biểu diễn tập mờ A như là sự kết nối giữa hai câu lệnh "x is near 3 AND y is near 4". Ở đây câu lệnh đầu tiên được định nghĩa: $\mu_{near3}\ x\ = g$aussian(x;3,2)

câu lệnh thứ hai được định nghĩa: $\mu_{near4}\ x\ = g$aussian(x;4,1)

Và tích giữa hai hàm thuộc trên được định nghĩa như là toán tử AND giữa câu lệnh.

Một loại hàm thuộc hai chiều khác là không kết hợp, ví dụ như tập mờ sau đây: $\mu_A\ x,y\ = \dfrac{1}{1+\left|x-3\right|\left|y-4\right|^{2.5}}$ thuộc loại không kết hợp.

1.1.3. Các phương pháp giải mờ

Vì việc xử lý kết hợp các tập mờ để tạo ra một tập mờ, việc này đồng nghĩa với việc đầu ra có một vùng giá trị, chính là tập nền và do vậy phải giải mờ để lấy một giá trị đầu ra từ tập mờ.

Nhiều kỹ thuật giải mờ đã được công bố nhưng thông dụng nhất vẫn là phương pháp trọng tâm (Centroid). Ngoài ra một số phương pháp khác như maxima, trung bình maxima, cao độ (heigh), cao độ cải tiến

1.1.3.1. Phương pháp giải mờ trọng tâm

Phương pháp này xác định trọng tâm y' của của vùng mờ B và đây chính là đầu ra của hệ logic mờ.

Với tập nền liên tục, phương pháp này như sau:

$$y' = \frac{\int_S y\mu_B(y)dy}{\int_S \mu_B(y)dy}$$

Trong đó S là miền xác định của $\mu_B \ y$

Với các biến rời rạc ta có công thức tính như sau:

$$y' = \frac{\sum_{i=1}^{N} y_i \mu_B(y_i)}{\sum_{i=1}^{N} \mu_B(y_i)}$$

Phương pháp giải mờ trọng tâm xác định điểm được cân bằng của vùng mờ kết quả bằng cách tính trung bình trọng số của các vùng mờ đầu ra. Đây là kỹ thuật được sử dụng rộng rãi nhất vì giá trị giải mờ có xu hướng dịch chuyển quanh vùng mờ đầu ra.

1.1.3.2. Phương pháp giải mờ maxima

Bộ giải mờ ước lượng tập mờ đầu vào và chọn giá trị giải mờ y sao cho $\mu_B \ y$ là cực đại. Không giống như phương pháp trọng tâm, phương pháp maxima chỉ được áp dụng vào một lớp hẹp các bài toán. Giá trị đầu ra của phương pháp này dễ bị thay đổi khi một luật có hàm thuộc hơn hẳn các luật khác. Vì vậy, kết quả có xu hướng nhảy từ khoảng này sang khoảng khác khi hình dạng vùng mờ thay đổi.

Phương pháp này tìm hai khoảng cao nhất, sau đó lấy điểm giữa của tâm hai khoảng này. Đó là kết quả đầu ra cần xác định.

1.1.3.3 Phương pháp giải mờ cao độ

Trước hết, bộ giải mờ tính $\mu_{B_i} \ y$ tại y_i, sau đó xác định đầu ra cho hệ logic mờ, với y_i là trọng tâm của tập mờ B_i. Đầu ra y_h được xác định:

$$y_h = \frac{\sum_{i=1}^{N} y_i ' \mu_B(y_i)}{\sum_{i=1}^{N} \mu_B(y_i)}$$

với m là số tập mờ đầu ra sau quá trình suy diễn. Phương pháp này dễ sử dụng và trọng tâm của các hàm thuộc mờ thông dụng được biết trước.

1.2. Mô hình hóa bài toán phân đoạn ảnh sử dụng phân cụm mờ

Phân đoạn ảnh giữ một vai trò rất quan trọng trong nhiều ứng dụng như các bài toán nhận dạng hay các bài toán xử lý ảnh. Phân đoạn ảnh là một bước cơ bản để có thể thực hiện việc phân tích các ảnh thu được. Một cách tổng quát, phân đoạn ảnh được định nghĩa như việc chia hình ảnh thành các đối tượng độc lập với nhau dựa trên các đặc tính của ảnh như mức xám hay kết cấu của ảnh. Có rất nhiều các thuật toán phân đoạn ảnh được đề xuất, chúng ta có thể chia ra làm 4 loại sau đây :

- Phương pháp cơ bản: phân ngưỡng, phát triển vùng, tách biên…
- Phương pháp thống kê: Maximum Likelihood Classifier (MLC)…
- Phương pháp dựa trên mạng Neural.
- Phương pháp dựa trên logic mờ (Fuzzy Clustering).

Chúng ta sẽ tập trung vào phương pháp trên logic mờ dựa trên mô hình phân cụm mờ Fuzzy C-means (FCM)

Phân lớp Fuzzy C-Means (FCM) là một trong những phương pháp được ứng dụng rộng rãi nhất trong Logic mờ. Được đưa ra bởi Bezdek bằng cách mở rộng thuật toán Dunn năm 1973, FCM là một trong những thuật toán hiệu quả trong bài toán phân lớp và đặc biệt là trong các bài toán phân đoạn ảnh. Với cách tiếp cận này, mỗi hình ảnh với nhiều đặc trưng sẽ được phân lớp thành các nhóm mà tại đó các điểm ảnh có cùng đặc trưng với nhau. Như vậy, bài toán phân lớp sẽ dẫn đến việc giải bài toán xác định giá trị min của tổng

khoảng cách của các điểm ảnh đến tâm của mỗi phân đoạn trên miền đặc trưng của ảnh.

Giả sử rằng X:= {x1, x2, ..., xn} định nghĩa tập các điểm ảnh của một ảnh cần phải phân thành c (0<c<n) phân đoạn {C1, C2, ..., Cc} trong đó $x_k \in R^d$ với k=1,2...n biểu diễn các đặc tính của điểm ảnh. Trong các ảnh thông thường, chúng ta thường hay xét đến giá trị mức xám, giá trị màu RGB của các điểm ảnh.

Xét ma trận phân lớp mờ (Fuzzy Partition Matrix) U = u_{ik} $_{cxn}$ trong đó mỗi phần tử u_{ik} chỉ ra khả năng thuộc phân lớp i của một điểm ảnh x_k. Khi đó, bài toán phân lớp chính là tối ưu hoá hàm mục tiêu:

$$J_m\ U,V\ = \sum_{k=1}^{n}\sum_{i=1}^{c} u_{ik}^m \left\|x_k - v_i\right\|^2$$

Trong đó ||.|| chính là giá trị chuẩn Euclidean trên không gian tương ứng và ma trận V biểu diễn tập hợp các điểm tâm của các phân lớp trong không gian này còn tham số m được gọi là tham số mờ của các tập dữ liệu. Khi đó, mô hình của bài toán phân đoạn ảnh được biểu diễn:

$$\begin{cases} J_m\ U,V\ = \sum_{k=1}^{n}\sum_{i=1}^{c} u_{ik}^m \left\|x_k - v_i\right\|^2 \\ u_{ik} \in [0,1], k = 1..n, i = 1..c, \sum_{i=1}^{c} u_{ik} = 1 \end{cases}$$

1.3. Kết luận

Trong chương này chúng ta đã nghiên cứu các vấn đề cơ bản về sự mở rộng của tập mờ loại một. Các phép toán cơ bản trên tập mờ loại một như các thuật toán tính trọng tâm, giải mờ…được giới thiệu để phục vụ việc thực hiện các thuật toán phân đoạn ảnh được trình bày trong các chương sau. Đồng thời, trong chương này cũng giới thiệu một cách khái quát mô hình hóa thuật toán phân đoạn ảnh sử dụng thuật toán FCM trong việc phân đoạn ảnh.

Chương 2
PHƯƠNG PHÁP PHÂN CỤM DỮ LIỆU

2.1. Khái niệm và mục tiêu của phân cụm dữ liệu

Phân cụm là một công cụ toán học dùng để phát hiện cấu trúc hoặc các mẫu nào đó trong tập dữ liệu, theo đó có đối tượng bên trong cụm dữ liệu thể hiện bậc tương đồng nhất định.

Nói cách khác, Phân cụm dữ liệu là quá trình nhóm một tập các đối tượng tương tự nhau trong tập dữ liệu vào các cụm sao cho các đối tượng thuộc cùng một cụm là tương đồng còn các đối tượng thuộc các cụm khác nhau sẽ không tương đồng.

Ngoài ra phân cụm dữ liệu còn có thể được sử dụng như một bước tiền xử lí cho các thuật toán khai phá dữ liệu khác như là phân loại và mô tả đặc điểm, có tác dụng trong việc phát hiện ra các cụm. Kỹ thuật phân cụm được áp dụng trong rất nhiều lĩnh vực như khai phá dữ liệu, nhận dạng mẫu, xử lý ảnh…

Với tư cách là một chức năng khai phá dữ liệu, phân tích phân cụm có thể được sử dụng như một công cụ độc lập chuẩn để quan sát đặc trưng của mỗi cụm thu được bên trong sự phân bố của dữ liệu và tập trung vào một tập riêng biệt của các cụm để giúp cho việc phân tích đạt kết quả.

Thuật toán phân cụm có nhiều dạng khác nhau từ phân cụm rõ đơn thuần như k-Means và phát triển đến thuật toán phân cụm mờ loại một Fuzzy c-Means (Bezdek, 1981) và gần đây là thuật toán phân cụm mờ loại hai khoảng - Interval Type II Fuzzy c-Means (Cheul Hwang và Frank Chung-Hoon Rhee, 2007).

Theo các nghiên cứu cho thấy thì hiện nay chưa có một phương pháp phân cụm tổng quát nào có thể giải quyết trọn vẹn cho tất cả các dạng cấu trúc

dữ liệu. Hơn nữa, các phương pháp phân cụm cần có cách thức biểu diễn cấu trúc của các dữ liệu, với mỗi cách thức biểu diễn khác nhau sẽ có tương ứng một thuật toán phân cụm phù hợp. Vì vậy phân cụm dữ liệu vẫn đang là một vấn đề khó và mở, vì phải giải quyết nhiều vấn đề cơ bản một cách trọn vẹn và phù hợp với nhiều dạng dữ liệu khác nhau, đặc biệt là đối với dữ liệu hỗn hợp đang ngày càng tăng trong các hệ quản trị dữ liệu và đây cũng là một trong những thách thức lớn trong lĩnh vực khai phá dữ liệu.

2.2. Những kỹ thuật cơ bản trong phân cụm dữ liệu

Các kỹ thuật phân cụm có rất nhiều cách tiếp cận và các ứng dụng trong thực tế, nó đều hướng tới hai mục tiêu chung đó là chất lượng của các cụm khám phá được và tốc độ thực hiện của thuật toán. Hiện nay, các kỹ thuật phân cụm có thể phân loại theo các cách tiếp cận chính sau :

2.2.1. Phương pháp phân cụm phân hoạch

2.2.1.1. Giới thiệu

Kỹ thuật này phân hoạch một tập hợp dữ liệu có n phần tử thành k nhóm cho đến khi xác định số các cụm được thiết lập. Số các cụm được thiết lập là các đặc trưng được lựa chọn trước. Phương pháp này là tốt cho việc tìm các cụm hình cầu trong không gian Euclidean. Ngoài ra, phương pháp này cũng phụ thuộc vào khoảng cách cơ bản giữa các điểm để lựa chọn các điểm dữ liệu nào có quan hệ là gần nhau với mỗi điểm khác và các điểm dữ liệu nào không có quan hệ hoặc có quan hệ là xa nhau so với mỗi điểm khác. Tuy nhiên, phương pháp này không thể xử lí các cụm có hình dạng kỳ quặc hoặc các cụm có mật độ các điểm dầy đặc. Các thuật toán phân hoạch dữ liệu có độ phức tạp rất lớn khi xác định nghiệm tối ưu toàn cục cho vấn đề phân cụm dữ liệu (PCDL), do nó phải tìm kiếm tất cả các cách phân hoạch có thể được. Chính vì vậy, trên thực tế thường đi tìm giải pháp tối ưu cục bộ cho vấn đề này bằng cách sử dụng một hàm tiêu chuẩn để đánh giá chất lượng của cụm

cũng như để hướng dẫn cho quá trình tìm kiếm phân hoạch dữ liệu. Như vậy, ý tưởng chính của thuật toán phân cụm phân hoạch tối ưu cục bộ là sử dụng chiến lược ăn tham để tìm kiếm nghiệm.

2.2.1.2. Thuật toán phân cụm tiêu biểu

Tiêu biểu cho các thuật toán phân cụm phân hoạch là thuật toán K-means Thuật toán này dựa trên độ đo khoảng cách của các đối tượng dữ liệu trong cụm. Trong thực tế, nó đo khoảng cách tới giá trị trung bình của các đối tượng dữ liệu trong cụm. Nó được xem như là trọng tâm của cụm. Như vậy, nó cần khởi tạo một tập trọng tâm các trọng tâm cụm ban đầu, và thông qua đó nó lặp lại các bước gồm gán mỗi đối tượng tới cụm mà trọng tâm gần, và tính toán tại tung tâm của mỗi cụm trên cơ sở gán mới cho các đối tượng. Quá trình lặp này dừng khi các trọng tâm hội tụ.

Mục đích của thuật toán K-means là sinh k cụm dữ liệu $\{C_1, C_2,..., C_k\}$ từ một tập dữ liệu chứa n đối tượng trong không gian d chiều $X_i = \{x_{i1}, x_{i2},..., x_{id}\}$, $i = 1 \div n$ sao cho hàm tiêu chuẩn:

$$E = \sum_{i=1}^{n} \sum_{x_i \in C_j} D^2 \ x_i - c_j$$

đạt giá trị tối thiểu. Trong đó: c_j là trọng tâm của cụm C_j, D là khoảng cách giữa hai đối tượng.

Mô tả bài toán:

Đầu vào: - Tập các đối tượng $X = \{X_i | i = 1, 2, \ldots, n\}$,

- Số cụm: k (1<k<n)

Đầu ra: Các cụm C_i ($i = 1 \div k$) tách rời và hàm tiêu chuẩn E đạt giá trị tối thiểu.

Thuật toán hoạt động trên một tập vectơ d chiều, tập dữ liệu X gồm n phần tử:

$$X = \{X_i \mid i = 1, 2, \ldots, n\}$$

$$X_i = \{x_{i1}, x_{i2},..., x_{id}\}, i = 1 \div n$$

Thuật toán K-means lặp lại nhiều lần quá trình:

- Tính toán khoảng cách.
- Cập nhật lại vị trí trọng tâm.

Quá trình lặp dừng lại khi trọng tâm hội tụ và mỗi đối tượng là nột bộ phận của một cụm.Hàm đo độ tương tự sử dụng khoảng cách Euclidean

$$E = \sum_{i=1}^{n} \sum_{x_i \in C_j} (\| x_i - c_j \|^2)$$

Trong đó cj là trọng tâm của cụm Cj

Hàm trên không âm, giảm khi có một sự thay đổi một trong hai bước: Tính toán khoảng cách và cập nhật vị trí trọng tâm.

Thuật toán K_means:

Bước 1 - Khởi tạo

Chọn k trọng tâm $c_i^{t=0}, i = 1 \div k$ ban đầu trong không gian R^d (d là số chiều của dữ liệu). Việc lựa chọn này có thể là ngẫu nhiên hoặc theo kinh nghiệm.

Bước 2 - Tính toán khoảng cách

Đối với mỗi điểm X_i $1 \le i \le n$, tính toán khoảng cách của nó với mỗi trọng tâm c_i $1 \le i \le k$. Sau đó tìm trọng tâm gần nhất với mỗi điểm và gán điểm vào tập S của trọng tâm gần nhất.

$$S_i^{(t)} = \left\{ \begin{matrix} x_j : \| x_j - c_i^{(t)} \| \le \| x_j - c_{i^*}^{(t)} \| \\ i^* = 1,.., k \end{matrix} \right\}$$

Bước 3 - Cập nhật lại trọng tâm:

Đối với mỗi $1 \le i \le k$, cập nhật trọng tâm cụm c_i bằng cách xác định trung bình cộng các vectơ đối tượng dữ liệu.

$$c_i^{(t+1)} = \frac{1}{|S_i^{(t)}|} \sum_{x_j \in S_i^{(t)}} x_j$$

Bước 4 – Điều kiện dừng

Lặp lại các bước 2 và 3 cho tới khi không có sự thay đổi trọng tâm của cụm.

Thuật toán K-means trên được chứng minh là hội tụ và có độ phức tạp tính toán là $O(3nkd\tau T^{flop})$. Trong đó, n là số đối tượng dữ liệu, k là số cụm dữ liệu, d là số chiều, τ là số vòng lặp, T^{flop} là thời gian để thực hiện một phép tính cơ sở như phép tính nhân, chia,... Như vậy, do K-means phân tích phân cụm đơn giản nên có thể áp dụng đối với tập dữ liệu lớn.Tuy nhiên, nhược điểm của K-means là chỉ áp dụng với dữ liệu có thuộc tính số và khám phá ra các cụm có dạng hình cầu, K-means còn rất nhạy cảm với nhiễu và các phần tử ngoại lai trong dữ liệu. Hơn nữa, chất lượng PCDL của thuật toán K-means phụ thuộc nhiều vào các tham số đầu vào như: số cụm k và k trọng tâm khởi tạo ban đầu. Trong trường hợp các trọng tâm khởi tạo ban đầu mà quá lệch so với các trọng tâm cụm tự nhiên thì kết quả phân cụm của K-means là rất thấp, nghĩa là các cụm dữ liệu được khám phá rất lệch so với các cụm trong thực tế.

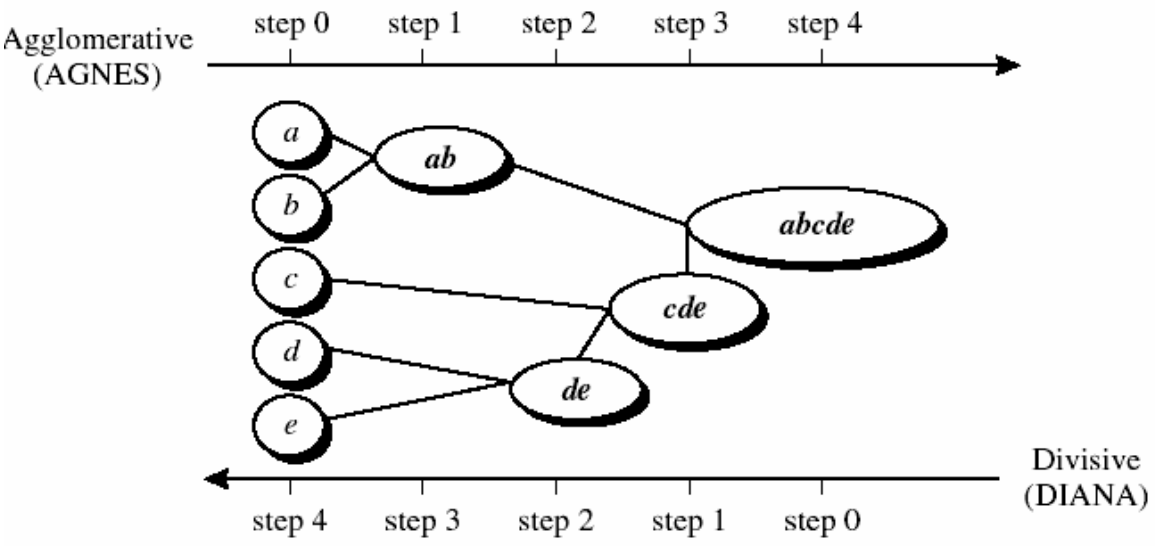
Trên thực tế chưa có một giải pháp tối ưu nào để chọn các tham số đầu vào, giải pháp thường được sử dụng nhất là thử nghiệm với các giá trị đầu vào khác nhau rồi sau đó chọn giải pháp tốt nhất.

2.2.2. Phương pháp phân cụm phân cấp

Phương pháp này xây dựng một phân cấp trên cơ sở các đối tượng dữ liệu đang xem xét. Nghĩa là sắp xếp một tập dữ liệu đã cho thành một cấu trúc có dạng hình cây, cây phân cấp này được xây dựng theo kỹ thuật đệ quy. Có hai cách tiếp cận phổ biến của kỹ thuật này đó là:

* Hòa nhập nhóm (Agglomerative), thường được gọi là tiếp cận Bottom-Up

* Phân chia nhóm (Divisive), thường được gọi là tiếp cận Top-Down



Hình 5 Chiến lược phân cụm phân cấp

Thực tế áp dụng, có nhiều trường hợp kết hợp cả hai phương pháp phân cụm phân hoạch và phân cụm phân cấp, nghĩa là kết quả thu được của phương pháp phân cấp có thể cải tiến thông qua bước phân cụm phân hoạch. Phân cụm phân hoạch và phân cụm phân cấp là hai phương pháp PCDL cổ điển, hiện đã có rất nhiều thuật toán cải tiến dựa trên hai phương pháp này đã được áp dụng phổ biến trong KPDL.

Tiêu biểu cho phương pháp phân cụm này là thuật toán CURE.Trong khi hầu hết các thuật toán thực hiện phân cụm với các cụm hình cầu và kích thước tương tự, như vậy là không hiệu quả khi xuất hiện các phần tử ngoại lai. Thuật toán CURE khắc phục được vấn đề này và tốt hơn với các phần tử ngoại lai. Thuật toán này định nghĩa một số cố định các điểm đại diện nằm rải rác trong toàn bộ không gian dữ liệu và được chọn để mô tả các cụm được hình thành. Các điểm này được tạo ra nhờ lựa chọn các đối tượng nằm rải rác cho cụm và sau đó "co lại" hoặc di chuyển chúng về trọng tâm cụm bằng nhân tố co cụm. Quá trình này được lặp lại và như vậy trong quá trình này, có

thể đo tỉ lệ gia tăng của cụm. Tại mỗi bước của thuật toán, hai cụm có cặp các điểm đại diện gần nhau (mỗi điểm trong cặp thuộc về mỗi cụm khác nhau) được hòa nhập.

Như vậy, có nhiều hơn một điểm đại diện mỗi cụm cho phép CURE khám phá được các cụm có hình dạng không phải là hình cầu. Việc co lại các cụm có tác dụng làm giảm tác động của các phần tử ngoại lai. Như vậy, thuật toán này có khả năng xử lí tốt trong trường hợp có các phần tử ngoại lai và làm cho nó hiệu quả với những hình dạng không phải là hình cầu và kích thước độ rộng biến đổi. Hơn nữa, nó tỉ lệ tốt với CSDL lớn mà không làm giảm chất lượng phân cụm.

Để xử lí được các CSDL 1ớn, CURE sử dụng mẫu ngẫu nhiên và phân hoạch, một mẫu là được xác định ngẫu nhiên trước khi được phân hoạch, và sau đó tiến hành phân cụm trên mỗi phân hoạch, như vậy mỗi phân hoạch là từng phần đã được phân cụm, các cụm thu được lại được phân cụm lần thứ hai để thu được các cụm con mong muốn, nhưng mẫu ngẫu nhiên không nhất thiết đưa ra một mô tả tốt cho toàn bộ tập dữ liệu.

Ngoài thuật toán CURE ra, phân cụm phân cấp còn bao gồm một số thuật toán khac như: Thuật toán BIRCH; Thuật toán AGNES (Agglomerative Nesting); Thuật toán DIANA (Divisive Analysis); Thuật toán ROCK; Thuật toán CHANMELEON.

2.2.3. Phương pháp phân cụm dựa trên mật độ

Kỹ thuật này nhóm các đối tượng dữ liệu dựa trên hàm mật độ xác định, mật độ là số các đối tượng lân cận của một đối tượng dữ liệu theo một nghĩa nào đó. Trong cách tiếp cận này, khi một dữ liệu đã xác định thì nó tiếp tục được phát triển thêm các đối tượng dữ liệu mới miễn là số các đối tượng lân cận này phải lớn hơn một ngưỡng đã được xác định trước. Phương pháp phân cụm dựa trên mật độ của các đối tượng để xác định các cụm dữ liệu có thể

phát hiện ra các cụm dữ liệu với hình thù bất kỳ. Kỹ thuật này có thể khắc phục được các phần tử ngoại lai hoặc giá trị nhiễu rất tốt, tuy nhiên việc xác định các tham số mật độ của thuật toán là rất khó khăn, trong khi các tham số này lại có tác động rất lớn đến kết quả phân cụm.

Tiêu biểu cho phương pháp phân cụm này là các thuật toán DBSCAN (Density-Based Spatial Clustering of Applications with Noise), thuật toán OPTICS (Density-Based Spatial Clustering of Applications with Noise), thuật toán DENCLUE (DENsity-based CLUstEring).

2.2.4. Phương pháp phân cụm dựa trên lưới

Kỹ thuật phân cụm dựa trên lưới thích hợp với dữ liệu nhiều chiều, dựa trên cấu trúc dữ liệu lưới để phân cụm, phương pháp này chủ yếu tập trung áp dụng cho lớp dữ liệu không gian. Mục tiêu của phương pháp này là lượng hóa dữ liệu thành các ô tạo thành cấu trúc dữ liệu lưới. Sau đó, các thao tác phân cụm chỉ cần làm việc với các đối tượng trong từng ô trên lưới chứ không phải các đối tượng dữ liệu. Cách tiếp cận dựa trên lưới này không di chuyển các đối tượng trong các ô mà xây dựng nhiều mức phân cấp của nhóm các đối tượng trong một ô. Phương pháp này gần giống với phương pháp phân cụm phân cấp nhưng chúng không trộn các ô, đồng thời giải quyết khắc phục yêu cầu đối với dữ liệu nhiều chiều mà phương pháp phân phân cụm dựa trên mật độ không giải quyết được. Ưu điểm của phương pháp phân cụm dựa trên lưới là thời gian xử lí nhanh và độc lập với số đối tượng dữ liệu trong tập dữ liệu ban đầu, thay vào đó là chúng phụ thuộc vào số ô trong mỗi chiều của không gian lưới.

Tiêu biểu cho phương pháp phân cụm này là các thuật toán STING (STatistical INformation Grid) (Wang, Yang và Munz 1997), thuật toán CLIQUE.

2.2.5. Phương pháp phân cụm dựa trên mô hình

Phương này cố gắng khám phá các phép xấp xỉ tốt của các tham số mô hình sao cho khớp với dữ liệu một cách tốt nhất. Chúng có thể sử dụng chiến lược phân cụm phân hoạch hoặc phân cụm phân cấp, dựa trên cấu trúc hoặc mô hình mà chúng giả định về tập dữ liệu và cách chúng hiệu chỉnh các mô hình này để nhận dạng ra các phân hoạch. Phương pháp phân cụm dựa trên mô hình cố gắng khớp giữa các dữ liệu với mô hình toán học, nó dựa trên giả định rằng dữ liệu được tạo ra bằng hỗn hợp phân phối xác suất cơ bản. Các thuật toán phân cụm dựa trên mô hình có hai cách tiếp cận chính: mô hình thống kê và mạng nơron. Phương pháp này gần giống với phương pháp phân cụm dựa trên mật độ, vì chúng phát triển các cụm riêng biệt nhằm cải tiến các mô hình đã được xác định trước đó, nhưng đôi khi nó không bắt đầu với một số cụm cố định và không sử dụng cùng một khái niệm mật độ cho các cụm.

Tiêu biểu cho phương pháp phân cụm này là thuật toán EM (Expectation-Maximization).Thuật toán EM được xem như là thuật toán dựa trên mô hình hoặc là mở rộng của thuật toán K-means. Thật vậy, EM gán các đối tượng cho các cụm đã cho theo xác suất phân phối thành phần của đối tượng đó. Phân phối xác suất thường được sử dụng là phân phối xác suất Gaussian với mục đích là khám phá lặp các giá trị tốt cho các tham số của nó bằng hàm tiêu chuẩn là hàm logarit khả năng của đối tượng dữ liệu, đây là hàm tốt để mô hình xác suất cho các đối tượng dữ liệu. EM có thể khám phá ra nhiều hình dạng cụm khác nhau, tuy nhiên do thời gian lặp của thuật toán khá nhiều nhằm xác định các tham số tốt nên chi phí tính toán của thuật toán khá cao. Đã có một số cải tiến được đề xuất cho EM dựa trên các tính chất của dữ liệu: có thể nén, có thể sao lưu trong bộ nhớ và có thể hủy bỏ. Trong các cải tiến này, các đối tượng bị hủy bỏ khi biết chắc chắn được nhãn phân cụm của nó, chúng được nén khi không bị loại bỏ và thuộc về một cụm quá

lớn so với bộ nhớ và chúng sẽ được lưu lại trong các trường hợp còn lại. Thuật toán được chia thành hai bước và quá trình đó được lặp lại cho đến khi vấn đề được giải quyết:

$$E:\mu \to a = \frac{\frac{1}{2}}{\frac{1}{2}+\mu} h, b = \frac{\mu}{\frac{1}{2}+\mu} h$$

$$M: a,b \to \mu = \frac{a+b}{6(b+c+d)}$$

Các bước thực hiện thuật toán EM

Khởi tạo tham số:

$$\lambda_0 = \ \mu_1^{\ 0}, \mu_2^{\ 0}, ..., \mu_k^{\ 0}, p_1^{\ 0}, p_2^{\ 0}, ..., p_k^{\ 0}$$

Bước E:

$$P\ \ \omega_j \mid x_k, \lambda_t\ = \frac{P\ \ \omega_j \mid x_k, \lambda_t\ \ P\ \ \omega_j, \lambda_t}{P\ \ x_k, \lambda_t} = \frac{P\ \ x_k \mid \omega_j, \lambda_i^{\ t}, \sigma^2\ \ P_i^{\ t}}{\sum_k P\ \ x_k \mid \omega_j, \lambda_j^{\ t}, \sigma^2\ \ P_j^{\ t}}$$

Bước M:

$$\mu_i^{\ t+1}\ = \frac{\sum_k P\ \ \omega_i \mid x_k, \lambda_t\ \ x_k}{\sum_k P\ \ \omega_i \mid x_k, \lambda_t}$$

$$p_i^{\ t+1}\ = \frac{\sum_k P\ \ \omega_i \mid x_k, \lambda_t}{R}$$

Lặp lại bước 2 và 3 cho đến khi đạt kết quả.

Ngoài thuật toán EM ra, phân cụm dựa trên mô hình còn có thêm một thuật toán khác là: Thuật toán COBWEB.

2.2.6. Phương pháp phân cụm có dữ liệu ràng buộc

Sự phát triển của PCDL không gian trên CSDL lớn đã cung cấp nhiều công cụ tiện lợi cho việc phân tích thông tin địa lí, tuy nhiên hầu hết các thuật toán này cung cấp rất ít cách thức cho người dùng để xác định các ràng buộc

trong thế giới thực cần phải được thỏa mãn trong quá trình phân cụm. Để PCDL không gian hiệu quả hơn, các nghiên cứu bổ sung cần được thực hiện để cung cấp cho người dùng khả năng kết hợp các ràng buộc trong thuật toán phân cụm.

Hiện nay, các phương pháp phân cụm trên đã và đang được phát triển và áp dụng nhiều trong các lĩnh vực khác nhau và đã có một số nhánh nghiên cứu được phát triển trên cơ sở của các phương pháp đó như:

Phân cụm thống kê: Dựa trên các khái niệm phân tích hệ thống, nhánh nghiên cứu này sử dụng các độ đo tương tự để phân hoạch các đối tượng, nhưng chúng chỉ áp dụng cho các dữ liệu có thuộc tính số.

Phân cụm khái niệm: Kỹ thuật này được phát triển áp dụng cho dữ liệu hạng mục, chúng phân cụm các đối tượng theo các khái niệm mà chúng xử lí. Phân cụm mờ: Sử đụng kỹ thuật mờ để PCDL. Các thuật toán thuộc loại này chỉ ra lược đồ phân cụm thích hợp với tất cả các hoạt động đời sống hàng ngày, chúng chỉ xử lí các dữ liệu thực không chắc chắn.

Phân cụm mạng Kohonen: Loại phân cụm này dựa trên khái niệm của các mạng nơron. Mạng Kohonen có tầng nơron vào và các tầng nơron ra. Mỗi nơron của tầng vào tương ứng với mỗi thuộc tính của bản ghi, mỗi một nơron vào kết nối với tất cả các nơron của tầng ra. Mỗi liên kết được gắn liền với một trọng số nhằm xác định vị trí của nơron ra tương ứng.

Phân cụm mờ: Fuzzy C-means (FCM), đồng phân cụm mờ (FCCI)

2.3. Kỹ thuật phân cụm dữ liệu mờ loại một

2.3.1. Tổng quan về phân cụm mờ

Trong cuộc sống, chúng ta đã gặp rất nhiều ứng dụng của bài toán phân cụm. Chẳng hạn như bài toán phân loại kết quả học tập trong nhà trường hay bài toán đưa thư trong ngành bưu điện… Đó chính là một ứng dụng của bài toán phân cụm rõ.

Ta có thể định nghĩa bài toán phân cụm rõ như sau: Cho tập dữ liệu mẫu X, ta kiểm tra các điểm dữ liệu xem nó giống với đặc điểm của nhóm nào nhất thì ta gán điểm dữ liệu đó vào trong nhóm đó.

Ví dụ 2.1: Phân loại kết quả học tập X học sinh

Kết quả học tập $DTB < 5.0$: Xếp loại học sinh yếu.

Kết quả học tập $5.0 \le DTB < 7.0$: Xếp loại học trung bình.

Kết quả học tập $7.0 \le DTB < 8.0$: Xếp loại học sinh khá.

Kết quả học tập $8.0 \le DTB < 9.0$: Xếp loại học sinh giỏi.

Kết quả học tập $9.0 \le DTB$: Xếp loại học sinh xuất xắc.

Trong đó DTB là điểm trung bình của học sinh.

Nhưng trong thực tế không phải lúc nào bài toán phân cụm rõ cũng áp dụng được.

Ví dụ 2.2: Phân loại người cao-thấp với tiêu chuẩn sau: những người cao ***khoảng 1.8m*** trở lên thì được xếp vào nhóm người cao ngược lại xếp vào nhóm người thấp. Vậy những người cao 1.799m thì ta xếp vào nhóm người nào?

Vì vậy, chúng ta cần đưa vào khái niệm bài toán phân cụm mờ. Trong các phương pháp phân cụm đã giới thiệu trong chương trước, mỗi phương pháp phân cụm phân hoạch một tập dữ liệu ban đầu thành các cụm dữ liệu có tính tự nhiên và mỗi đối tượng dữ liệu chỉ thuộc về một cụm dữ liệu, phương pháp này chỉ phù hợp với việc khám phá ra các cụm có mật độ cao và rời nhau, với đường biên giữa các cụm được xác định tốt. Tuy nhiên, trong thực tế, đường biên giữa các cụm có thể mờ, các cụm có thể chồng lên nhau, nghĩa là một số các đối tượng dữ liệu thuộc về nhiều các cụm khác nhau, do đó mô hình này không mô tả được dữ liệu thực. Vì vậy người ta đã áp dụng lý thuyết về tập mờ trong PCDL để giải quyết cho trường hợp này. Cách thức kết hợp này được gọi là Phân cụm mờ. Phân cụm mờ là phương pháp phân cụm dữ

liệu mà cho phép mỗi điểm dữ liệu thuộc về hai hoặc nhiều cụm thông qua bậc thành viên. Ruspini (1969) giới thiệu khái niệm phân hoạch mờ để mô tả cấu trúc cụm của tập dữ liệu và đề xuất một thuật toán để tính toán tối ưu phân hoạch mờ. Dunn (1973) mở rộng phương pháp phân cụm và đã phát triển thuật toán phân cụm mờ. Ý tưởng của thuật toán là xây đựng một phương pháp phân cụm mờ dựa trên tối thiểu hóa hàm mục tiêu. Bezdek (1981) cải tiến và tổng quát hóa hàm mục tiêu mờ bằng cách đưa ra trọng số mũ để xây dựng thuật toán phân cụm mờ và được chứng minh độ hội tụ của các thuật toán là cực tiểu cục bộ.

2.3.2. Thuật toán Fuzzy C-means (FCM)

2.3.2.1. Giới thiệu thuật toán

Nếu như K-means là thuật toán PCDL rõ thì FCM là thuật toán phân cụm mờ tương ứng, hai thuật toán này cùng sử dụng chung một chiến lược phân cụm dữ liệu. Thuật toán FCM đã được áp dụng thành công trong giải quyết một số lớn các bài toán PCDL như trong nhận dạng mẫu (nhận dạng vân tay, ảnh), xử lý ảnh (phân tách các cụm ảnh màu, cụm màu), y học (phân loại bệnh, phân loại triệu chứng), … Tuy nhiên, nhược điểm lớn nhất của thuật toán FCM là tập dữ liệu lớn, tập dữ liệu nhiều chiều, nhạy cảm với các nhiễu và phần tử ngoại lai trong dữ liệu, nghĩa là các trung tâm cụm có thể nằm xa so với trọng tâm thực của cụm.

Kỹ thuật này phân hoạch một tập n vectơ đối tượng dữ liệu $X = x_1, x_2, ..., x_n \in R^d$ thành c các nhóm mờ dựa trên tính toán tối thiểu hóa hàm mục tiêu để đo chất lượng của phân hoạch và tìm trọng tâm cụm trong mỗi nhóm, sao cho chi phí hàm đo độ phi tương tự là nhỏ nhất. Một phân hoạch mờ vectơ điểm dữ liệu $X = x_1, x_2, ..., x_n \in R^d$ là đặc trưng đầu vào

được biểu diễn bởi ma trận $U = u_{ik}$ sao cho điểm dữ liệu đã cho chỉ có thể thuộc về một số nhóm với bậc được xác định bởi mức độ thuộc giữa [0, 1].

Như vậy, ma trận U được sử dụng để mô tả cấu trúc cụm của X bằng cách giải thích u_{ik} như bậc thành viên x_k với cụm i.

Cho $u = u_1, u_2, ..., u_c$ là phân hoạch mờ C

$$U_{cxn} = \begin{pmatrix} u_{11} & \cdots & u_{1n} \\ \vdots & \ddots & \vdots \\ u_{c1} & \cdots & u_{cn} \end{pmatrix}$$

Dunn định nghĩa hàm mục tiêu mờ như sau:

$$J_m(U,v) = \sum_{k=1}^{n}\sum_{i=1}^{c} u_{ik}(d_{ik})^2$$

Bezdek khái quát hóa hàm mục tiêu mờ bằng cách đưa ra trọng số mũ $m > 1$, là số thực nào đó bất kỳ như sau :

$$J_m(U,v) = \sum_{k=1}^{n}\sum_{i=1}^{c} u_{ik}{}^{m}(d_{ik})^2, \qquad 1 \le m \le \infty \tag{2.1}$$

Trong đó :

$X = x_1, x_2, ..., x_n \in R^d$ là nửa dưới vector mẫu dữ liệu tập con thực d chiều trong không gian vector R^d.

$m \in 1, +\infty$ là trọng số mũ hay còn gọi là tham số mờ.

$v_i \in R^d$ là trọng tâm của cụm thứ i

$d_{ik} = d(x_k - v_i) = \|x_k - v_i\| = \left[\sum_{j=1}^{d}(x_{kj} - v_{ij})^2\right]^{1/2}$ là khoảng cách theo thước đo Euclide giữa mẫu dữ liệu x_k với trọng tâm cụm thứ i , v_i

$u_{ik} \in 0,1$ là bậc hay độ thuộc của dữ liễu mẫu x_k với cụm thứ i

$V = \left[v_{ji}\right] = v_1, ..., v_c \in R^{dxc}$ là ma trận biểu diễn các giá trị tâm của cụm

Để thuận tiện, coi mảng đối tượng dữ liệu $x_1,....,x_n$ là các cột trong ma trận đối tượng dữ liệu $X=\left[x_{jk}\right]=x_1,...,x_n \in R^{dxc}$. Ma trận phân hoạch U được sử dụng để mô tả cấu trúc cụm trong dữ liệu $x_1,....,x_n$.

Định nghĩa 2.1: Họ các tập mờ $(u_{A_i},A_i),i=1,2,...,c = \underset{\sim i}{A},i=1,2,...,c$ trong không gian vũ trụ $X = x_1,x_2,...,x_n$ được gọi là phân hoạch mờ của X nếu bậc của dữ liệu mẫu thỏa mãn điều kiện :

$$\begin{cases} 0 \le u_{ik} \le 1, & 1 \le i \le c, 1 \le k \le n \\ 0 < \sum_{k=1}^{n} u_{ik} < n, & 1 \le i \le c \\ \sum_{i=1}^{c} u_{ik} = 1, & 1 \le k \le n \end{cases} \qquad (2.2)$$

Dễ nhận thấy: $\underset{\sim i}{A} \cap \underset{\sim j}{A} \neq \varnothing$ tức là $Min(u_{ik},u_{jk}) > 0$

Như vậy mỗi phân hoạch mờ cũng có biểu diễn bằng một ma trận c hàng và n cột để biểu diễn phân hoạch n đối tượng thành c cụm dữ liệu trong không gian R^{cxn} được viết gọn như sau :

$$M_{fcn} = \left\{ U \in R^{cxn} \mid \forall i,k : u_{ik} \in [0,1] ; \sum_{i=1}^{c} u_{ik} = 1; 0 < \sum_{k=1}^{n} u_{ik} < n \right\} \qquad (2.3)$$

R^{cxn} là không gian của tất cả các ma trận thực cấp $c \times n$.

Tập M_{fc} có thể là tập vô hạn, tức là ta không thể xây dựng được công thức tính số phương án phân hoạch $\eta_{M_{fc}} (\eta_{M_{fc}} = \infty)$

Thông thường ta gọi bài toán phân cụm mờ là bài toán tìm các độ thuộc u_{ij} nhằm tối thiểu hóa hàm mục tiêu (2.1) với các điều kiện sau :

Định lý 2.1: Nếu m và c là các tham số cố định, và I_k là một tập được định nghĩa như sau:

$$\underset{1\le k\le n}{\forall} I_k = \{i \mid 1 \le i \le c, d_{ik} = 0\} \tag{2.4}$$

Thì hàm mục tiêu (2.1) đạt min khi và chỉ khi:

$$u_{ik} = \begin{cases} \dfrac{1}{\sum_{j=1}^{c}\left(\dfrac{d_{ik}}{d_{jk}}\right)^{\frac{2}{m-1}}}, I_k = \varnothing \\ \begin{cases} 0, i \notin I_k \\ \sum_{i\in I_k} u_{ik} = 1, i \in I_k \; I_k \neq \varnothing \end{cases} \end{cases} \quad ,1 \le i \le c, 1 \le k \le n \tag{2.5}$$

$$v_i = \frac{\sum_{k=1}^{n}(u_{ik})^m x_k}{\sum_{k=1}^{n}(u_{ik})^m}, 1 \le i \le c \tag{2.6}$$

Định lý này đã được Bezdek chứng minh là đúng nếu $m \ge 1, d_{ik}^2 > 0, 1 \le i \le c$

Một phân hoạch tối ưu, nghĩa là hàm mục tiêu (2.1) đạt giá trị tối thiểu, mà chủ yếu dựa trên đó độ tương tự giữa x_k và trọng tâm cụm v_i, điều này tương đương với hai điều kiện (2.5) và (2.6) phải thỏa mãn các ràng buộc.

2.3.2.2. Thuật toán FCM

Thuật toán FCM cung cấp một quá trình lặp qua lại giữa phương trình (2.5) và (2.6) để xấp xỉ cực tiểu hàm mục tiêu (2.1) dựa trên độ đo tương tự có trọng số giữa x_k và trọng tâm cụm v_i, sau mỗi vòng lặp, thuật toán tính toán và cập nhật các phần tử u trong ma trận phân hoạch U. Phép lặp sẽ dừng khi $\max \left\| u_{ij}^{k+1} - u_{ij}^{k} \right\| \le \varepsilon$ trong đó ε là chuẩn kết thúc nằm trong khoảng $0,1$ trong khi k là các bước lặp. Thủ tục này hội tụ tới cực tiểu cục bộ hay điểm yên ngựa của $J_m\ u,V$. Thuật toán FCM tính toán ma trận phân hoạch U và

kích thước của các cụm để thu được các mô hình mờ từ ma trận này. Các bước thực hiện của thuật toán FCM như sau:

Input: Số cụm *c* và tham số mũ *m* cho hàm mục tiêu J, sai số ε

Output: c cụm dữ liệu sao cho hàm mục tiêu (2.1) đạt giá trị cực tiểu

Begin

Bước 1. Khởi tạo

Nhập tham số *c (1<c<n), m(1<m<+∞),* ε

Khởi tạo ma trận $V = \left[v_{ij} \right], V^{(0)} \in R^{d \times c}, j = 0$

Bước 2. Tính ma trận phân hoạch U và cập nhật lại trọng tâm cụm V

2.1. j=j+1

2.2. Tính ma trận phân hoạch mờ U^j theo công thức (2.5)

2.3 Cập nhật các trọng tâm cụm $V^{(j)} = \left[v_1^{(j)}, v_2^{(j)}, ..., v_c^{(j)} \right]$ theo công thức (2. 6) và U^j

Bước 3: Kiểm tra điều kiện dừng. Nếu $max \left\| u_{ij}^{k+1} - u_{ij}^{k} \right\| \leq \varepsilon$ chuyển sang bước 4, ngược lại quay lại bước 2.

Bước 4. Đưa ra các cụm kết quả.

<u>Nhận xét</u>*:*

Việc chọn các tham số cụm rất ảnh hưởng đến kết quả phân cụm, tham số này thường được chọn theo phương pháp ngẫu nhiên hoặc theo Heuristic.

Đối với $m \to 1^+$ thì thuật toán FCM trở thành thuật toán rõ

Đối với $m \to \infty$ thì thuật toán FCM trở thành thuật toán phân cụm mờ với $u_{ik} = \frac{1}{c}$. Chưa có quy tắc nào nhằm chọn lựa tham số m đảm bảo cho phân cụm hiệu quả, thông thường chọn m=2.

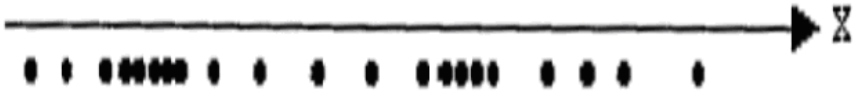
Ta có thể tiến hành đánh giá việc lựa chọn số tâm cụm tối ưu:

$$\min_{(c)}\left\{P(c)=\sum_{i=1}^{c}\sum_{k=1}^{n}u_{ik}^{m}(\left\|\mathbf{x}_k-\mathbf{v}_i\right\|^2-\left\|\mathbf{v}_i-\bar{\mathbf{x}}\right\|^2)\right\}$$

Trong đó: $\bar{\mathbf{x}}=\frac{1}{n}\sum_{k=1}^{n}\mathbf{x}_k$

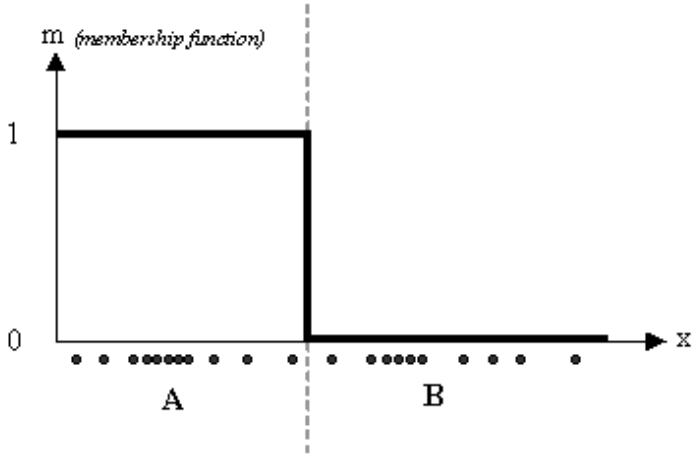
2.3.2.3. So sánh FCM với K-means

Để so sánh có thể xét ví dụ sau: Cho một tập các đối tượng dữ liệu một chiều được biểu thị như hình 2.2 dưới đây.



Hình 6 Mô tả tập dữ liệu một chiều

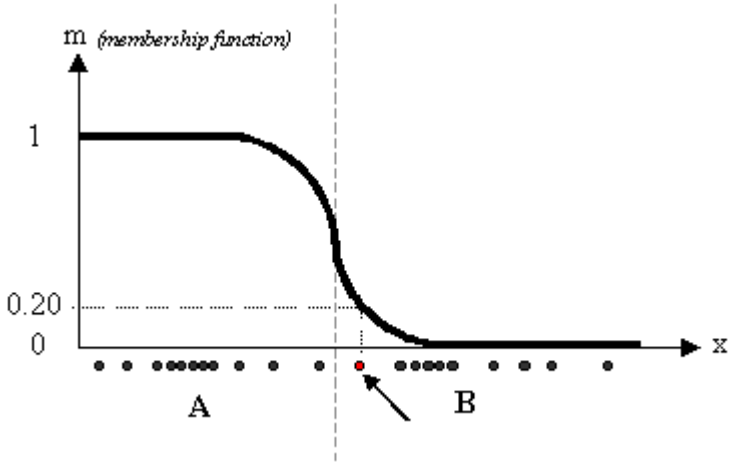
Bằng quan sát dễ nhận thấy có hai cụm trong tập dữ liệu trên đặt tên tương ứng là "A" và "B". Với thuật toán K-means thì hàm tính độ phụ thuộc giữa đối tượng dữ liệu và trọng tâm cụm của nó được thể hiện như trong đồ thị hình 2.3 dưới đây:



Hình 7 Hàm thuộc với trọng tâm của cụm A trong K-means

Dựa vào hình rút ra nhận xét rằng, các đối tượng trong cụm A có giá trị hàm thuộc với trọng tâm của cụm A là bằng 1 và bằng 0 với trọng tâm cụm B. Điều này ngược lại với các đối tượng ở trong cụm B.

Đối với thuật toán FCM thì hàm thuộc của các đối tượng dữ liệu với các trọng tâm cụm dữ liệu được minh họa như trong đồ thị hình 2.4 dưới đây:

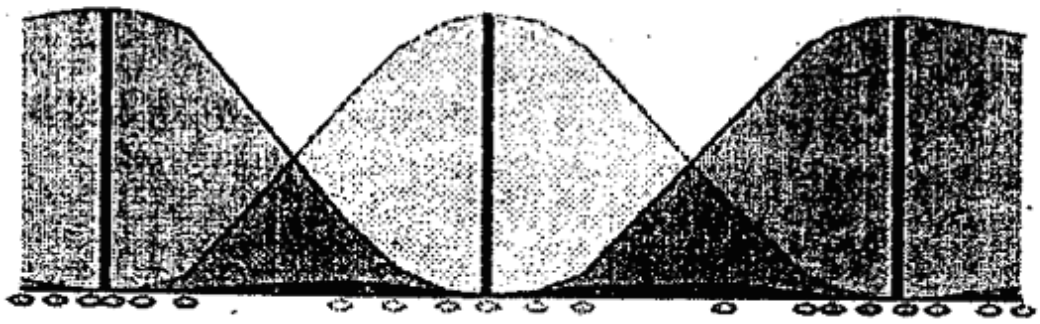


Hình 8 Hàm thuộc với trọng tâm cụm A trong FCM

Dựa vào hình có thể nhận xét rằng, các đối tượng dữ liệu có giá trị hàm thuộc với các trọng tâm của cụm A nằm trong khoảng [0, l], hàm thuộc lúc này là một đường cong trơn. Điểm có mũi tên chỉ đến có nhiều khả năng thuộc về lớp B hơn là lớp A do giá trị hàm thuộc của nó vào lớp A là nhỏ (=0.2). Có thể biểu diễn các giá trị hàm thuộc trên bằng ma trận cho cả hai trường hợp như sau:

$$U_{n\times c} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ \dots & \\ 1 & 0 \\ & \end{pmatrix} \text{ và } U_{n\times c} = \begin{pmatrix} 0.8 & 0.2 \\ 0.3 & 0.7 \\ \dots & \\ 0.9 & 0.1 \\ & \end{pmatrix}$$

Số dòng và số cột phụ thuộc vào số các đối tượng dữ liệu n và số các cụm k. Một số ví dụ mô phỏng về kết quả các cụm khám phá được của thuật toán phân cụm mờ FCM như hình 2.5 dưới đây.



Hình 9 Các cụm được khám phá bởi thuật toán FCM

Độ phức tạp của thuật toán FCM tương đương với độ phức tạp của thuật toán K-means trong trường hợp số đối tượng của tập dữ liệu cần phân cụm là rất lớn.

Tóm lại, thuật toán phân cụm mờ FCM là một mở rộng của thuật toán K-means nhằm để khám phá ra các cụm chồng lên nhau. Tuy nhiên, FCM vẫn chứa đựng các nhược điểm của thuật toán K-means trong việc xử lý đối với các phần tử ngoại lai và nhiễu trong dữ liệu. Thuật toán εFCM được trình bày phần sau là một mở rộng của thuật toán FCM nhằm khắc phục các nhược điểm này.

2.3.3. Thuật toán FCM cải tiến.

2.3.3.1. Cơ sở thuật toán.

Ta có thể tăng tốc độ tính toán của thuật toán FCM bằng cách giảm các phép toán thực hiện. Từ công thức (2.6) ta biết rằng mỗi tâm cụm v_i được tính bằng trung bình của các mẫu dữ liệu trong cụm thứ i. Trong thuật toán FCM chuẩn v_i được tính toán bằng cách duyệt qua toàn bộ tập dữ liệu và ma trận phân hoạch.

Tiếp theo đây chúng tôi mô tả thuật toán cải tiến mà ở đó việc tính các tâm cụm được thực hiện theo trình tự cập nhật ma trận phân hoạch. Hiệu quả

của sự cải tiến này loại trừ một lần duyệt qua toàn bộ tập dữ liệu trong mỗi lần lặp. Kết quả là không giảm số lượng các vòng lặp được yêu cầu để hội tụ, nhưng giảm thời gian trong mỗi lần lặp.

Ta duy trì hai cấu trúc mở rộng, một ma trận $c \times n$, $P = p_i{}_{c \times n}$ và một véc tơ q có chiều dài là c. Giá trị khởi tạo ban đầu của P và q được lấy từ tử số và mẫu số của công thức (2.6) khi ma trận thành viên phân hoạch được tạo ra. Lúc này, công thức (2.6) được viết lại như sau:

$$v_i = p_i / q_i \qquad (2.7)$$

với p_i là một véc tơ có chiều dài n, và q_i là một giá trị vô hướng.

Mỗi lần một phần tử u_{ki} của ma trận thành viên được so sánh là một lần tử số của công thức (2.6) tăng lên, p_i tăng lên một lượng:

$$((u_{ik}^{(j+1)})^m - (u_{ik}^{(j)})^m)x_k \qquad (2.8)$$

và mẫu số của công thức một cũng tăng, q_i tăng lên một lượng:

$$(u_{ik}^{(j+1)})^m - (u_{ik}^{(j)}) \qquad (2.9)$$

Những gia tăng này được tích lũy vào P và q theo thứ tự ma trận phân hoạch được cập nhật. Bắt đầu vòng lặp tiếp theo các tâm chùm mới được tính lại theo công thức (2.7).

2.3.3.2. Thuật toán FCM cải tiến

Thuật toán FCM cải tiến thực hiện các bước như sau:

Input: Số cụm c và tham số mũ m cho hàm mục tiêu J, sai số ε

Output: c cụm dữ liệu sao cho hàm mục tiêu (2.1) đạt giá trị cực tiểu

Begin

Bước 1. Khởi tạo

Nhập tham số $c\ (1<c<n)$, $m(1<m<+\infty)$, ε

Khởi tạo ma trận $V = \left[v_{ij} \right], V^{(0)} \in R^{d \times c}, j = 0$

Khởi tạo các cấu trúc dữ liệu P và q sử dụng công thức (2.6).

Bước 2. Tính ma trận phân hoạch U và cập nhật lại trọng tâm cụm V

2.1. j=j+1

2.2. Tính ma trận phân hoạch mờ U^{j} theo công thức (2.5) và tăng các phần tử tương ứng trong P và q dùng công thức (2.8) và (2.9)

2.3 Cập nhật các trọng tâm cụm $V^{(j)} = \left[v_1^{(j)}, v_2^{(j)}, ..., v_c^{(j)} \right]$ theo công thức (2.7)

Bước 3: Kiểm tra điều kiện dừng. Nếu $max \left\| u_{ij}^{k+1} - u_{ij}^{k} \right\| \leq \varepsilon$ chuyển sang bước 4, ngược lại quay lại bước 2.

Bước 4. Đưa ra các cụm kết quả.

End.

2.3.4. Thuật toán ε- Insensitive Fuzzy C-means (εFCM)

2.3.4.1. Giới thiệu thuật toán

Thuật toán phân cụm FCM sử dụng hàm bậc hai để đo độ phi tương tự giữa dữ liệu và các trọng tâm cụm. Suy luận sử dụng độ do này là tính toán thấp và đơn giản. Tuy nhiên, cách tiếp cận này dễ bị ảnh hưởng bởi nhiễu và các phần tử ngoại lai. Để khắc phục nhược điểm trên, một độ đo cải tiến đã được đề xuất (Vapnik, 1998) sử dụng tham số ε như sau :

$$\left\| t \right\|_{\varepsilon} = \begin{cases} 0, & \left\| t \right\| < \varepsilon \\ \left\| t \right\| - \varepsilon, & \left\| t \right\| > \varepsilon \end{cases}, \ \varepsilon \text{ là tham số nhạy cảm với nhiễu.}$$

Hàm mục tiêu của thuật toán εFCM được định nghĩa như sau:

$$J_{m\varepsilon}\ U,V\ = \sum_{k=1}^{n} \sum_{i=1}^{c}\ u_{ik}\ ^{m} \left\| d_{ik} \right\|_{\varepsilon} \qquad (2.10)$$

Trong đó

$$\left\|d_{ik}\right\|_{\varepsilon} = \left\|x_k - v_i\right\|_{\varepsilon} = \sum_{l=1}^{d} \left\|x_{kl} - v_{il}\right\|_{\varepsilon} \tag{2.11}$$

Ta cố định V và U là độc lập, cực tiểu hàm mục tiêu (2.10) có thể biểu diễn như sau:

$$J_{m\varepsilon}\ U,V\ = \sum_{k=1}^{n} g_k\ U \tag{2.12}$$

$$\text{Trong đó } \underset{1\le k\le n}{\forall} g_k\ U\ = \sum_{i=1}^{c}\ u_{ik}\ ^{m} \left|x_k - v_i\right|_{\varepsilon} \tag{2.13}$$

Khai triển Lagrange (2.12) với ràng buộc (2.3) ta được:

$$\underset{1\le k\le n}{\forall} G_k\ U,\lambda\ = \sum_{i=1}^{c}\ u_{ik}\ ^{m} \left|x_k - v_i\right|_{\varepsilon} - \lambda\left[\sum_{i=1}^{c} u_{ik} - 1\right] \tag{2.14}$$

Trong đó λ là hệ số Lagrange.

Thiết lập đạo hàm Lagrange bằng không, ta có:

$$\underset{1\le k\le n}{\forall} \frac{\partial G_k\ U,\lambda}{\partial \lambda} = \sum_{i=1}^{c} u_{ik} - 1 = 0 \tag{2.15}$$

Và

$$\underset{\substack{1\le s\le c\\ 1\le k\le n}}{\forall} \frac{\partial G_k\ U,\lambda}{\partial u_{sk}} = m\ u_{sk}\ ^{m-1} \left|x_k - v_s\right|_{\varepsilon} - \lambda = 0 \tag{2.16}$$

Từ (2.16) ta có:

$$u_{sk} = \left(\frac{\lambda}{m}\right)^{\frac{1}{m-1}}\ \left|x_k - v_s\right|_{\varepsilon}\ ^{\frac{1}{1-m}} \tag{2.17}$$

Từ (2.17) và (2.15) ta có:

$$\left(\frac{\lambda}{m}\right)^{\frac{1}{m-1}} \sum_{j=1}^{c}\ \left|x_k - v_j\right|_{\varepsilon}\ ^{\frac{1}{1-m}} = 1 \tag{2.18}$$

Kết hợp (2.17) và (2.18) ta được:

$$\underset{\substack{1\le s\le c\\ 1\le k\le n}}{\forall} u_{sk} = \left|x_k - v_s\right|_\varepsilon^{\frac{1}{1-m}} \Bigg/ \left[\sum_{j=1}^{c} \left|x_k - v_j\right|_\varepsilon^{\frac{1}{1-m}}\right] \tag{2.19}$$

Định nghĩa 2.2: Tập I_k được định nghĩa là:

$$I_k = \{ i \mid 1 \le i \le c; \|d_{ik}\|_\varepsilon = 0 \ , k = 1,2.....,N \}$$

Định lý 2.2: Nếu m, c và ε là các tham số xác định, với $U,V \in M_{fc} \times R_{dc}$ (2.10) đạt cực tiểu cho hàm mục tiêu $J_{m\varepsilon}(U,V)$ nếu:

$$\underset{\substack{1\le i\le c\\ 1\le k\le N}}{\forall} u_{ik} = \begin{cases} \|x_k - v_i\|_\varepsilon^{\frac{1}{1-m}} \left[\sum_{j=1}^{c} \|x_k - v_j\|_\varepsilon^{\frac{1}{1-m}}\right]^{-1}, I_k = \varnothing \\ \begin{cases} 0, i \notin I_k \\ \sum_{i\in I_k} u_{ik} = 1, i \in I_k \end{cases} I_k \ne \varnothing \end{cases} \tag{2.20}$$

Vấn đề đặt ra bây giờ là cập nhật trọng tâm cụm. Kết hợp (2.10) và (2.11) ta được:

$$J_{m\varepsilon}(U,V) = \sum_{i=1}^{c}\sum_{k=1}^{n} (u_{ik})^m \sum_{l=1}^{d} \left|x_{kl} - v_{il}\right|_\varepsilon = \sum_{i=1}^{c}\sum_{l=1}^{d} g_{il}(v_{il}) \tag{2.21}$$

Trong đó $g_{il}(v_{il}) = \sum_{k=1}^{n} (u_{ik})^m \left|x_{kl} - v_{il}\right|_\varepsilon$ (2.22)

Ta sử dụng các biến phụ $\xi_k^+, \xi_k^- \ge 0$ để tất cả các x_{kl} ta có thể viết:

$$\begin{cases} x_{kl} - v_{il} \le \varepsilon + \xi_k^- \\ v_{il} - x_{kl} \le \varepsilon + \xi_k^+ \end{cases} \tag{2.23}$$

Biểu thức (2.22) có thể viết lại như sau:

$$g_{il}(v_{il}) = \sum_{k=1}^{n} (u_{ik})^m (\xi_k^+ + \xi_k^-) \tag{2.24}$$

Khai triển Langrange (2.24) với ràng buộc (2.23) ta được:

$$G_{il}\ v_{il}\ = \sum_{k=1}^{n} u_{ik}\ ^{m}\ \xi_k^+ + \xi_k^-\ - \sum_{k=1}^{n} \lambda_k^+\ \varepsilon + \xi_k^+ - v_{il} + x_{kl} \\ - \sum_{k=1}^{n} \lambda_k^-\ \varepsilon + \xi_k^- + v_{il} - x_{kl}\ - \sum_{k=1}^{n}\ \mu_k^+ \xi_k^+ + \mu_k^- \xi_k^- \tag{2.25}$$

Trong đó $\lambda_k^+, \lambda_k^-, \mu_k^+, \mu_k^- \geq 0$ là các hệ số Langrange.

Để có thể cực tiểu (2.12) thì cực đại các hệ số Langrange, chính vì vậy phải thỏa mãn điều kiện:

$$\begin{cases} \dfrac{\partial G_{il}\ v_{il}}{\partial v_{il}} = \sum_{k=1}^{n}\ \lambda_k^+ - \lambda_k^-\ = 0 \\ \dfrac{\partial G_{il}\ v_{il}}{\partial \xi_k^+} =\ u_{ik}\ ^{m} - \lambda_k^+ - \mu_k^+ = 0 \\ \dfrac{\partial G_{il}\ v_{il}}{\partial \xi_k^-} =\ u_{ik}\ ^{m} - \lambda_k^- - \mu_k^- = 0 \end{cases} \tag{2.26}$$

Từ hai điều kiện cuối của (2.26) và $\mu_k^+, \mu_k^- \geq 0$ ta có kết quả: $\lambda_k^+, \lambda_k^- \in \left[0,\ u_{ik}\ ^{m}\right]$. Kết hợp điều kiện (2.26) vào khai triển Lagrange (2.25) ta có :

$$G_{il}\ v_{il}\ = -\sum_{k=1}^{n}\ \lambda_k^+ - \lambda_k^-\ - \varepsilon \sum_{k=1}^{n}\ \lambda_k^+ + \lambda_k^- \tag{2.27}$$

Để cực đại (2.27) thì ta có:

$$\begin{cases} \sum_{k=1}^{n}\ \lambda_k^+ - \lambda_k^-\ = 0 \\ \lambda_k^+, \lambda_k^- \in \left[0,\ u_{ik}\ ^{m}\right] \end{cases} \tag{2.28}$$

Và đây chính là công thức song song Wolfe thỏa mãn điều kiện:

$$\begin{cases} \lambda_k^+ \ \varepsilon + \xi_k^+ - v_{il} + x_{kl} = 0 \\ \lambda_k^- \ \varepsilon + \xi_k^- + v_{il} - x_{kl} = 0 \\ u_{ik}{}^m - \lambda_k^+ \ \xi_k^+ = 0 \\ u_{ik}{}^m - \lambda_k^- \ \xi_k^- = 0 \end{cases} \tag{2.29}$$

Từ hai điều kiện cuối của (2.29) ta thấy rằng $\lambda_k^+, \lambda_k^- \in 0,\ u_{ik}{}^m \Rightarrow \xi_k^+, \xi_k^- = 0.$

Trong trường hợp này từ hai điều kiện đầu (2.29) ta có:

$$\underset{1\le i\le c}{\forall} \underset{1\le l\le d}{\forall} v_{il} = \begin{cases} x_{kl} + \varepsilon \ , & \underset{k|\lambda_k^+ \in 0,\ u_{ik}{}^m}{\forall} \\ x_{kl} - \varepsilon \ , & \underset{k|\lambda_k^- \in 0,\ u_{ik}{}^m}{\forall} \end{cases} \tag{2.30}$$

Vì vậy chúng ta cập nhật trọng tâm cụm v_{il} theo công thức (2.30) bằng việc lấy các dữ liệu x_{kl} mà các hệ số khai triển Langrange $\lambda_k^+, \lambda_k^- \in 0,\ u_{ik}{}^m$. Tuy nhiên, ta sẽ cập nhật trọng tâm cụm v_{il} bằng cách lấy trung bình các giá trị v_{il} đạt được từ tất cả các dữ liệu x_{kl} thỏa mãn điều kiện (2.30).

2.3.4.2. Chi tiết thuật toán εFCM

Các bước thực hiện thuật toán εFCM như sau:

Input : Số cụm c (1<c<n) và tham số $m \in 1,\infty\ , \varepsilon$ cho hàm mục tiêu J

Output: Các cụm dữ liệu sao cho hàm mục tiêu trong (2.10) đạt giá trị cực tiểu

Begin

Bước 1. Khởi tạo.

Nhập tham số *c (1<c<n), m(1<m<+∞),* ε, ξ

Khởi tạo ma trận $V^{\ 0} = \left[v_{ij} \right], V^{(0)} \in R^{d\times c}, j = 0$

Bước 2. Thực hiện tính ma trận phân hoạc U và cập nhật trọng tâm cụm V

2.1. j=j+1

2.2. Tính ma trận phân hoạch mờ U^{j} theo công thức (2.20)

2.3 Cập nhật các trọng tâm cụm $V^{(j)} = \left[v_1^{(j)}, v_2^{(j)}, ..., v_c^{(j)} \right]$ theo công thức (2.27) , (2.30) và U^{j}

Bước 3. Kiểm tra điều kiện dừng: nếu $\max \| U^{(j+1)} - U^{(j)} \| \leq \xi$ thì chuyển sang bước 4, ngược lại quay lại bước 2.

Bước 4. Đưa ra các cụm kết quả.

End.

Tóm lại, thuật toán ε FCM là một mở rộng của thuật toán FCM trong việc thích nghi với nhiễu và phần tử ngoại lai trong dữ liệu. Tuy vậy, hiệu quả của thuật toán εFCM đối với tập dữ liệu lớn, tập dữ liệu nhiều chiều cũng như cách xác định tham số ε là những vấn đề tiếp tục cần phải nghiên cứu và hoàn thiện.

Chương 3
PHƯƠNG PHÁP ĐỒNG PHÂN CỤM MỜ

3.1.Tiến triển của thuật toán và các công việc liên quan

Đồng phân cụm đồng thời phân đoạn theo cả đối tượng và theo đặc trưng. Vì vậy nên cần có 2 hàm thuộc : hàm đối tượng (hay được gọi là hàm phân vùng) và hàm của đặc trưng(hàm xếp hạng). Hàm xếp hạng phục vụ cho việc lọc ra các đặc trưng liên quan với nhau trong thời gian tính của hàm thuộc đối tượng. Thuật toán đồng phân cụm được chứng minh là phù hợp với những dữ liệu nhiều chiều và đã được kiểm chứng trên ảnh đa màu sắc.Vấn đề về phần tử ngoại lai cũng được tối thiểu hóa bằng cách sử dụng hàm đặc trưng. Vấn đề với việc chỉ sử dụng hàm thuộc đặc trưng là nó có thể dẫn đến trùng/chồng chéo cụm, do đó việc sử dụng cả hàm thuộc của đặc trưng và đối tượng là điều cần thiết.Ngoài ra hàm tính toán khoảng cách giữa các điểm đặc trưng và các tâm cụm được đưa thêm vào trong quá trình xử lý đồng phân cụm để tăng thêm hiệu quả so với các thuật toán đồng phân cụm khác . Đồng phân cụm được kết hợp với một số thuật toán đánh giá khác để đưa ra được số lượng cụm chính xác nhất . Cả 2 hàm đặc trưng và đối tượng đều mờ,ví dụ như hàm đối tượng được tính toán khi các cụm khác nhau cạnh tranh 1 điểm dữ liệu , và hàm đặc trưng được xét đến khi các đặc trưng khác nhau cạnh tranh nhau 1 cụm. Như vậy chúng ta sẽ có sự kết hợp giữa 2 hàm thuộc mờ (đối tượng và đặc trưng) trong phương pháp này.

Do đó mục đích là để có một thuật toán đồng phân cụm với những ưu điểm sau:

1. Ít nhạy cảm với thứ tự dữ liệu đầu vào
2. Thực hiện tốt trong tập dữ liệu nhiều chiều và xác định các cụm được rõ

3. Hạn chế tối đa tác động của các giá trị ngoại lai để cải thiện tính chính xác của đồng phân cụm(bằng cách sử dụng 2 hàm thuộc đặc trưng/xếp hạng)
4. Thuật toán phải đủ nhanh

Một cách tiếp cận là biến thể của thuật toán FCM nhưng tính toán theo entropy.Nó liên quan đến việc tối thiểu hàm mục tiêu sau đây:

$$J_{FCM} = \sum_{c=1}^{C}\sum_{i=1}^{N} u_{ci} Dist(x_i,p_c) + T_U \sum_{c=1}^{C}\sum_{i=1}^{N} u_{ci} \log u_{ci} \qquad (1)$$

Với điều kiện

$$\sum_{c=1}^{C} u_{ci} = 1,\ u_{ci} \in [0,1], \forall i = 1,...,N \qquad (2)$$

Với C,N là số cụm và số điểm dữ liệu , u_{ci} là hàm thuộc mờ, T_u là trọng số , $Dist(x_i,p_c)$ là bình phương khoảng cách Euclidean giữa điểm x_i và tâm cụm p_c. Biểu thức thứ nhất của (1) biểu thị khoảng cách giữa điểm và tâm cụm,phần thứ hai đóng vai trò như phần điều chỉnh trong quá trình tối thiểu hàm. Trước hết,chúng ta hãy thử áp dụng phương pháp đồng phân cụm cho thuật toán FCM này bằng cách thay thế hàm tính toán khoảng cách $Dist(x_i,p_c)$ bằng $Dist(x_{ij},p_{cj})$ trong hàm mục tiêu,trong đó $Dist(x_{ij},p_{cj})$ được tính toán cho từng đặc trưng tách biệt j=1,2,…,K .

3.2. Xây dựng hàm mục tiêu

Đặt $X=\{ x_1, x_2, \ldots, x_i, \ldots, x_N\} \in R^k$ là N điểm của 1 ảnh I có kích cỡ $N_1 x N_2 = N$, và K là số chiều của của không gian đặc trưng gắn liền với mỗi điểm dữ liệu. Gọi x_{ij} là đặc trưng thứ j của điểm thứ i, $P = \{p_{cj}\}$ là tập hợp các tâm cụm của đặc trưng thứ j và $D_{cij} = Dist(x_{ij},p_{cj})$ là bình phương khoản cách Euclidean giữa điểm x_{ij} và tâm cụm p_{cj} tính bằng :

$$D_{cij} = d^2(x_{ij}, p_{cj}) = (x_{ij} - p_{cj})^2 \qquad (3)$$

Đặt u_{ci} là hàm đối tượng của điểm thứ i cụm c, U = $\{u_{ci}\}$ là ma trận CxN của hàm đối tượng của ảnh I , v_{cj} là hàm đặc trưng của đặc trứng j với tâm cụm thứ c và V = $\{v_{cj}\}$ sẽ là ma trân CxK của hàm đặc trưng của ảnh I.

Thêm hàm thuộc đặc trưng v_{cj} vào phần 1 của (1) và thay thế hàm khoảng cách bởi D_{cij} = Dist(x_{ij}, p_{cj}) ta có $\sum_{c=1}^{C}\sum_{i=1}^{N}\sum_{j=1}^{K} u_{ci}v_{cj}D_{cij}$ .Biểu thức entropy $\sum_{c=1}^{C}\sum_{i=1}^{N} u_{ci}\log u_{ci}$ và $\sum_{c=1}^{C}\sum_{j=1}^{K} v_{cj}\log v_{cj}$ của hàm đối tượng và hàm đặc trưng sẽ đóng vai trò là phần thứ 2 và 3 cấu tạo lên hàm mục tiêu J_{FCCI} . Tối thiểu hóa 2 hàm này tương đương với tối đa hóa entropy mờ của $-\sum_{c=1}^{C}\sum_{i=1}^{N} u_{ci}\log u_{ci}$ và $-\sum_{c=1}^{C}\sum_{j=1}^{K} v_{cj}\log v_{cj}$. Trong đó các hàm thuộc mờ u_{ci} và v_{cj} đều có ràng buộc.

Hàm mục tiêu J_{FCCI} kết quả từ việc kết hợp tất cả các vấn đề trên là :

$$J_{FCCI}(\mathbf{U},\mathbf{V},\mathbf{P}) = \sum_{c=1}^{C}\sum_{i=1}^{N}\sum_{j=1}^{K} u_{ci}v_{cj}D_{cij} + T_U \sum_{c=1}^{C}\sum_{i=1}^{N} u_{ci}\log u_{ci} + T_V \sum_{c=1}^{C}\sum_{j=1}^{K} v_{cj}\log v_{cj} \quad (4)$$

Với ràng buộc

$$\sum_{c=1}^{C} u_{ci} = 1,\ u_{ci}\in[0,1],\forall i = 1,\ldots,N \quad (5)$$

$$\sum_{j=1}^{K} v_{cj} = 1,\ v_{cj}\in[0,1],\ \forall c = 1,\ldots,C \quad (6)$$

Tối thiểu hóa phần đầu của công thức (4) gán các đối tượng có giá trị của hàm thuộc cao hơn vào cụm mà nó gần hơn. Thành phần bên trong $\{v_{cj}D_{cij}\}$ gán các trọng số cao hơn vào những tính năng nổi bật và trọng số thấp hơn với những tính năng còn lại . T_U và T_V là các trọng số mờ. Tăng T_U Và T_V sẽ làm tăng sự mờ của các cụm lên.

3.3. Cải tiến phương trình

Để tối ưu hóa hàm mục tiêu FCCI từ (4) có thể áp dụng hệ số nhân Lagrange λi và γc cho các ràng buộc (5) và (6) tương ứng như sau.

$$J(\mathbf{U},\mathbf{V},\mathbf{P}) = \sum_{c=1}^{C}\sum_{i=1}^{N}\sum_{j=1}^{K} u_{ci}v_{cj}D_{cij} + T_U \sum_{c=1}^{C}\sum_{i=1}^{N} u_{ci}\log u_{ci} + T_V \sum_{c=1}^{C}\sum_{j=1}^{K} v_{cj}\log v_{cj} + \sum_{i=1}^{N}\lambda_i\left(\sum_{c=1}^{C} u_{ci}-1\right) + \sum_{c=1}^{C}\gamma_c\left(\sum_{j=1}^{K} v_{cj}-1\right) \quad (7)$$

Lấy đạo hàm riêng của J(U,V,P) trong công thức (7) đối với U và đặt gradient bằng 0 thì ta sẽ có,

$$\frac{\partial J}{\partial \mathrm{U}} = \sum_{j=1}^{K} v_{cj}D_{cij} + T_U(1+\log u_{ci}) + \lambda_i = 0 \quad (8)$$

Thay thế u_{ci} tính được trong (8) vào (5) ta có công thức tính hàm thuộc của đối tượng u_{ci},

$$u_{ci} = \frac{e^{\left(-\sum_{j=1}^{K}\frac{v_{cj}D_{cij}}{T_U}\right)}}{\sum_{c=1}^{C} e^{\left(-\sum_{j=1}^{K}\frac{v_{cj}D_{cij}}{T_U}\right)}} \quad (9)$$

Tương tự như vậy,tính toán đạo hàm riêng của J(0,V,P) theo V và đặt gradient bằng 0 thì ta có,

$$\frac{\partial J}{\partial \mathrm{V}} = \sum_{i=1}^{N} u_{ci}D_{cij} + T_V(1+\log v_{cj}) + \gamma_c = 0 \quad (10)$$

Tính v_{cj} từ (10) và áp dụng vào công thức (6) ta có công thức tính hàm thuộc đặc trưng v_{cj} như sau

$$v_{cj} = \frac{e^{\left(-\sum_{i=1}^{N}\frac{u_{ci}D_{cij}}{T_V}\right)}}{\sum_{j=1}^{K} e^{\left(-\sum_{i=1}^{N}\frac{u_{ci}D_{cij}}{T_V}\right)}} \quad (11)$$

Lấy đạo hàm riêng của J(U,V,P) theo P và đặt gradient bằng 0 thì

$$\frac{\partial J}{\partial P} = v_{cj} \sum_{i=1}^{N} u_{ci} x_{ij} - v_{cj} p_{cj} \sum_{i=1}^{N} u_{ci} = 0 \qquad (12)$$

Từ (12) ta tính được p_{cj} như sau :

$$p_{cj} = \frac{\sum_{i=1}^{N} u_{ci} x_{ij}}{\sum_{i=1}^{N} u_{ci}} \qquad (13)$$

Các công thức (9), (11) và (13) là những phương trình cập nhật cho các hàm đối tượng,hàm đặc trưng và các tâm cụm trong mỗi lần lặp.Để xác định số tâm cụm chính xác ta sử dụng thêm thuật toán Xie-Beni.

3.4. Giã mã cho thuật toán FCCI

1. Xác định trọng số T_U, T_V, độ chính xác giải thuật ε ,số tâm cụm C và số lần lặp lớn nhất τ_{max}.
2. Đặt số lặp $\tau = 1$.
3. Xác định u_{ci} với $0 \leq u_{ci} \leq 1$.
4. Lặp
5. Tính p_{cj} sử dụng công thức (13)
6. Tính D_{cij} sử dụng công thức (3).
7. Tính v_{cj} sử dụng công thức (11).
8. Tính u_{ci} sử dụng công thức (9).
9. Xét lại $\tau = \tau + 1$.
10. Dừng lại cho đến khi $\max(|u_{ci}(\tau) - u_{ci}(\tau - 1)|) \leq \varepsilon$ hoặc $\tau = \tau_{max}$.

Với kết quả của nhiều thử nghiệm thì thường ta đặt $\tau_{max} = 200$ và $\varepsilon = 10^{-2}$.

3.5. Phân cụm ảnh màu sử dụng FCCI

3.5.1. Đánh giá giá trị của cụm theo thuật toán Xie và Beni

Theo Xie và Beni, công thức tính giá trị S của các cụm trong trường hợp tối nhất là

$$S = \frac{\sigma / N}{d_{min}^{\ 2}} \qquad (14)$$

Giá trị d_{min} trong (14) được tính như sau :

$$d_{min} = \min_{\forall c}\{\sum_{j=1}^{K}(p_{(c+1)j} - p_{cj})^2\} \quad (15)$$

khi d_{min} là khoảng cách nhỏ nhất giữa tâm cụm p_{cj} của cụm c =1,…,C và đặc trưng j và σ là là sự thay đổi lớn nhất trong số tất cả các cụm C, được tính bằng :

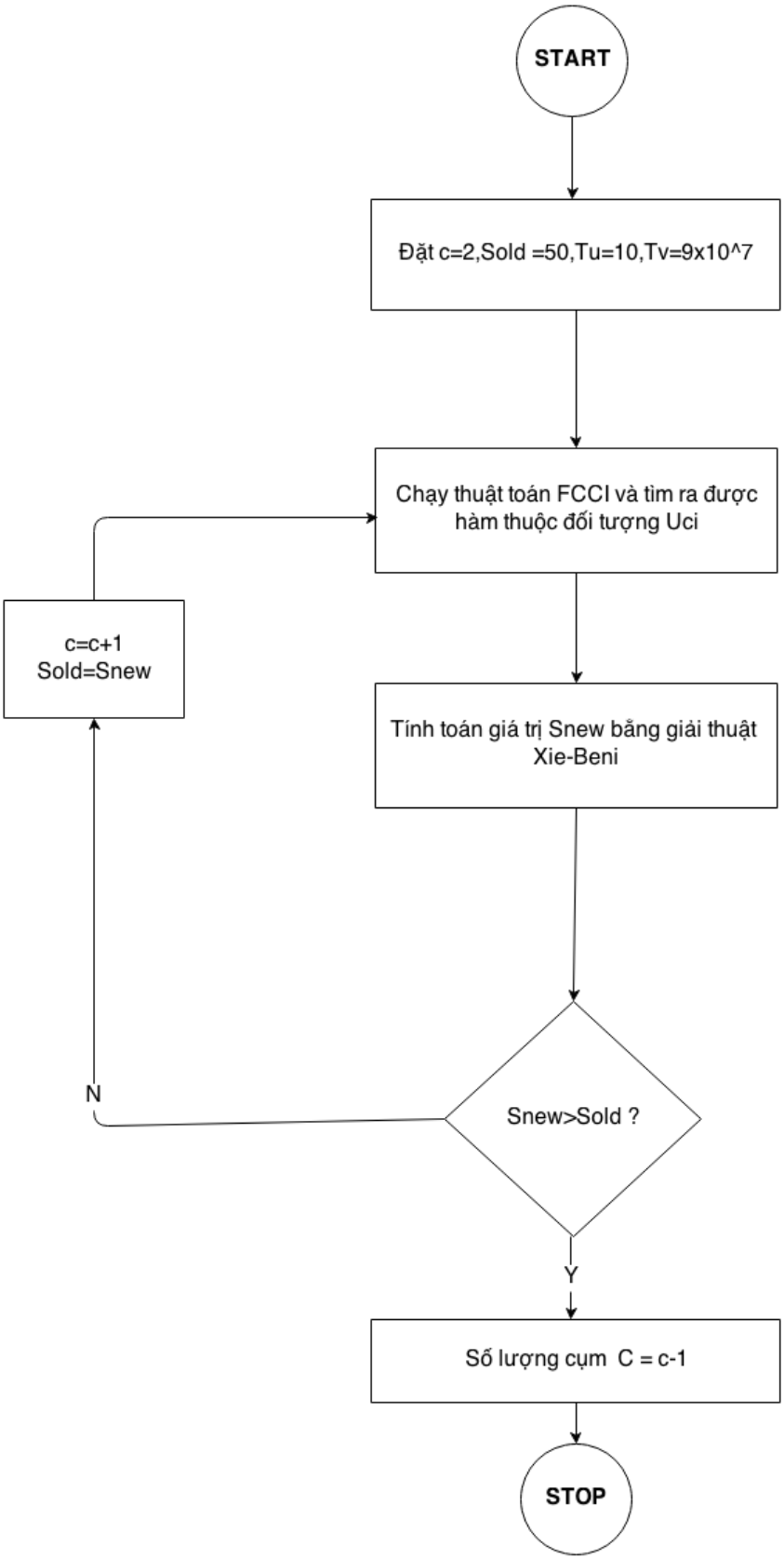
$$\sigma = \max_{\forall c}\{\sum_{i=1}^{N} u_{ci}{}^2 \sum_{j=1}^{K}(x_{ij} - p_{cj})^2\} \quad (16)$$

Để xác định được số lượng cụm dựa trên giá trị S trên,ta sử dụng thuật toán như hình 3.1 ,thuật toán này sẽ kiểm tra sự xuất hiện của cực tiểu địa phương đầu tiên của S.

3.5.2. Thuật toán phân cụm ảnh màu sử dụng FCCI

Thuật toán gồm cách bước sau :

1. Lấy dữ liệu từ ảnh 3 chiều RBG đầu vào.
2. Chuyển đổi hệ màu RGB sang hệ màu CIELAB với số chiều K=2, ví dụ. {a^*,b^*}. Không gian L*a*b* bao gồm một '* L' lớp sáng, lớp kết tủa màu 'a*' và lớp kết tủa màu 'b*'. Tất cả các thông tin màu sắc nằm trong các lớp ' a*' và 'b *'.
3. Chuyển đổi dữ liệu 2 chiều sang dữ liệu 1 chiều để tạo ra các điểm x_{ij} trong chiều j, j=1,2 với mỗi điểm ảnh i=1,…,N, với N là độ lớn của dữ liệu.Bước này khá quan trọng vì việc tính toán trên mảng dữ liệu 1 chiều thì đơn giản hơn trên mảng dữ liệu 2 chiều.
4. Tính toán số cụm C và chạy thuật toan FCCI dựa trên sơ đồ như hình 3.1
5. Sau khi tính toán xong thuật toán, giải mờ u_{ci} vào các cụm



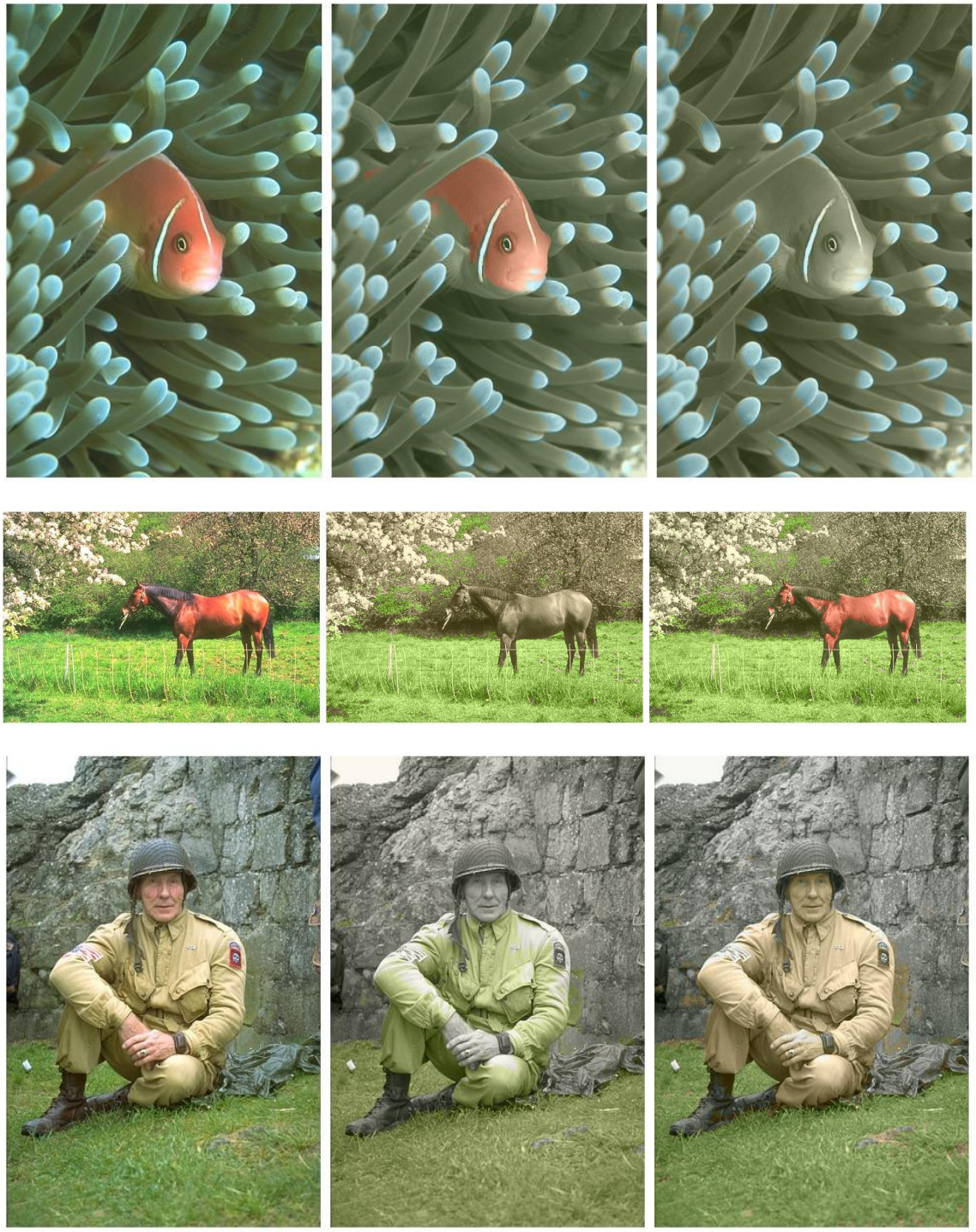
Hình 10 Giải thuật tìm số cụm của ảnh

3.5.3. Kết quả của thuật toán FCCI

Chạy thuật toán FCCI trên tập dữ liệu ảnh lấy từ cơ sở dữ liệu ảnh của đại học Berkeley,với mỗi ảnh có kích thước 321 x 481 hoặc 481 x 321 thì thời gian trung bình để chạy là mất tầm 50s và số bước lặp trung bình là 33 với số lượng cụm phát hiện ra < 7 cụm.

Một số kết quả của thuật toán được liệt kê ra như sau :

Theo thứ tự từ trái qua phải : ảnh gốc,ảnh với 2 cụm,ảnh với 3 cụm (ảnh với 4 cụm)…



Hình 11 Kết quả của thuật toán FCCI trên ảnh màu

KẾT LUẬN VÀ KIẾN NGHỊ

1. Kết luận

Với đề tài "Nghiên cứu thuật toán đồng phân cụm mờ cho bài toàn phân đoạn ảnh", tôi nhận thấy lĩnh vực phân cụm dữ liệu và logic mờ là hết sức rộng l ảnh.

Chính vì vậy, luận văn chỉ tập trung tìm hiểu, nghiên cứu và trình bày được một số kỹ thuật và thuật toán phân cụm dữ liệu phổ biến như K-means và FCM hay FCM cải tiến. Ngoài ra, trong luận văn này còn trình bày những thuật toán rất mới hiện nay là thuật toán đồng phân cụm FCCI và tiến hành cài đặt thử nghiệm thuật toán FCCI trong bài toán phân đoạn ảnh.

Thông thường các thuật toán phân cụm thường gặp khó khăn trong việc khởi tạo tâm cụm ban đầu, cũng như việc xác định số tâm cụm một cách chính xác. Điều này làm cho việc sử dụng các thuật toán phân cụm hoặc là không ổn định hoặc đạt kết quả không mong muốn.

Luận văn đã đề xuất thuật toán phân cụm FCCI kết hợp với thuật toán Xie-Beni nhằm thực hiện phân cụm chính xác bằng cách khởi tạo số lượng tâm cụm ban đầu sau đó tiến hành tính toán giá trị theo công thức Xie-Beni và tăng số lượng cụm lên 1 cho đến khi gặp cực tiểu địa phương đầu tiên. Thuật toán thực hiện phân đoạn trên ảnh cho kết quả tốt, ổn định.

2. Kiến nghị

Do thời gian và điều kiện nghiên cứu còn hạn chế, nên kết quả nghiên cứu mới dừng lại ở mô phỏng phân đoạn trên các ảnh màu mà chưa ứng dụng vào hệ thống thực tế để góp phần nâng cao hiệu quả, chất lượng của hệ thống.

Trên đây là toàn bộ bản thuyết minh luận văn thạc sỹ kỹ thuật của tác giả. Trong quá trình thực hiện không tránh khỏi những thiếu sót, tác giả rất mong nhận được sự đóng góp ý kiến của thầy cô, bạn bè và đồng nghiệp.

TÀI LIỆU THAM KHẢO

1. Madasu Hanmandlu, Om Prakash Verma, Seba Susan.*'Color segmentation by fuzzy co-clustering of chrominance color features*' – Neurocomputing - Volume 120, 23 November 2013, Pages 235–249