THUẬT TOÁN PHÂN CỤM SUBTRACTIVE VÀ ỨNG DỤNG PHÂN ĐOẠN ẢNH MÀU

I. GIỚI THIỆU

Phân cụm trừ là quá trình chia tập dữ liệu và các cụm thoả mãn các đối tượng trong một cụm tương tự nhau. Còn các đối tượng khác cụm thì khác nhau. Phân cụm đã được ứng dụng rộng rãi trong các lĩnh vực phân loại thị trường, phân loại khách hàng, nhận dạng mẫu, phân tích dữ liệu không gian, phân loại dữ liệu web. … Có nhiều thuật toán phân loại dữ liệu đã được phát triển như k-means, k-medoids, c-means mờ, mountian, phân cụm trừ.

Các thuật toán k-means và c-means đã được ứng dụng rộng rãi. Cả 2 thuật toán này bắt đầu với việc khởi tạo số lượng cụm C và các tâm của các cụm ban đầu x_1, x_2, …, x_c. Hiệu quả phân cụm dựa vào việc lựa chọn các giá trị ban đầu này. Vấn đề đặt ra là xác định giá trị ban đầu đó để thuật toán đạt hiệu quả nhất.

Yager và Filev đã đưa ra một thuật toán đơn giản và hiệu quả là giải thuật mountain để xác định số lượng cụm và tâm các cụm ban đầu. Phương pháp này đưa ra một không gian lưới và đánh giá khả năng trở thành tâm cụm của các điểm lưới dựa và khoảng cách tới các điểm dữ liệu thực. Giải thuật mountain tuy đơn giản nhưng tính toán phức tạp vì lên tới hàm mũ.

Chiu đã đưa ra một cải tiến của giải thuật mountain đó là thuật toán phân cụm trừ (subtractive clustering). Thuật toán này đuược xây dựng dựa trên thuật toán mountain với việc đưa ra hàm tính mật độ để tính toán khả năng trở thành tâm cụm cho từng điểm dữ liệu dựa vào khoảng cách của điểm dữ liệu này với tất cả các điểm dữ liệu còn lại. Giảu thuật này chỉ xem xét đến các điểm dữ liệu mà không cần xét đến các điểm lưới lân cận điểm dữ liệu, chính điều này làm cho giải thuật trở nên đơn giản hơn so với giải thuật mountain và tốc độ tính toán được cải thiện hơn.

II. THUẬT TOÁN PHÂN CỤM TRỪ.

Thuật toán phân cụm trừ, đánh giá khả năng trở thành tâm cụm của các điểm dữ liệu dựa vào mật độ của các điểm lân cận, để tính mật độ cho từng điểm dữ liệu. SC đưa ra một giá trị r gọi là bán kín cụm. Điểm dữ liệu có mật độ lớn nhất sẽ trở thành tâm cụm.

Giả sử tập dữ liệu có n điểm $\{x_1, x_2, \ldots..x_n\}$, mật độ ban đầu của mỗi điểm dữ liệu được tính như sau:

$$P_i = \sum_{j=1}^{n} e^{-\alpha \left\| x_i - x_j \right\|^2} \qquad (1)$$

Trong đó: $\alpha = \frac{4}{r_a^2}$, r_a là bán kính cụm, P_i là mật độ của điểm dữ liệu thứ i, n là số điểm dữ liệu, $\|x_i - x_j\|$ là khoảng cách Euclid giữa điểm dữ liệu thứ i với điểm dữ liệu thứ j. Lựa chọn điểm có mật độ lớn nhất làm tâm cụm thứ nhất. Gọi x_1^* là vị trí tâm cụm đầu tiên, có mật độ là P_1^* với $P_1^* = \max_{i=1}^{n} P_i$. Giả sử đã tìm được tâm cụm thứ k là x_k^* có mật độ là P_k^*. Khi đó mật độ của các điểm còn lại được tính như sau:

$$P_i = P_i - P_k^* e^{-\beta \|x_i - x_k^*\|^2} \quad (2)$$

Trong đó: $\beta = \frac{4}{r_b^2}$, $r_b = \eta * r_a$, hệ số η là một hằng số lớn hơn 1, P_i là mật độ của điểm thứ i. Hằng số r_b là một giá trị lớn hơn r_a để tâm cụm tiếp theo sẽ nằm ngoài phạm vị r_a so với tâm cụm thứ k.

Để tìm tâm cụm thứ (k+1), tìm x* có mật độ lớn nhất P* trong các điểm còn lại và đánh giá khả năng trở thành tâm cụm của điểm x*. Thuật toán sử dụng 2 hằng số gọi là hằng số chấp nhận $\bar{\varepsilon}$ và hằng số từ chối $\underline{\varepsilon}$ để đánh giá trở thành tâm cụm mới của x*.

Nếu P*> $\bar{\varepsilon} * P_k^*$ thì x* là tâm cụm thứ (k+1) tìm được. Nếu P*< $\underline{\varepsilon} * P_k^*$ thì thuật toán dừng và kết quả thu được k tâm cụm.

Nếu $\underline{\varepsilon} * P_k^*$ <= P*<= $\bar{\varepsilon} * P_k^*$, thì xét giá trị của biểu thức

$$F = \frac{d_{\min}}{r_a} + \frac{p^*}{P_k^*}$$

với $F = \frac{d_{\min}}{r_a} + \frac{p^*}{P_k^*}$ là khoảng cách nhỏ nhất giữa điểm x* tới tâm cụm trước đó, nếu F>=1 thì x* là tâm cụm thứ (k+1), ngược lại thiết lập P*=0 và lặp việc tìm tâm cụm thứ (k+1).

Khi kết thúc thuật toán, để đưa ra kết quả là các cụm dữ liệu xét độ phụ thuộc của từng điểm dữ liệu vào các tâm cụm. Độ phụ thuộc của điểm dữ liệu xi vào tâm cụm thứ k được tính theo công thức:

$$\mu_{ik} = e^{-\alpha \|x_i - x_k^*\|^2} \quad (3)$$

III. CÁC BƯỚC THUẬT TOÁN

Bước 0: Tính mật độ ban đầu cho các điểm dữ liệu theo công thức (1)

Bước 1: Chọn điểm x_1^* có mật độ lớn nhất $P_1^* = \max_{i=1}^{n} P_i$ làm tâm cụm thứ nhất. Tính lại mật độ cho các điểm còn lại theo công thức (2)

Bước k: (k>1): đã tìm được tâm cụm thứ k là x_k^* có mật độ là P_k^*. Tìm x* là điểm có mật độ lớn nhất P*.

- Nếu P*> $\bar{\varepsilon} * P_k^*$: Thuật toán dừng, tìm được k tâm cụm
- Ngược lại:

  + Nếu $F = \frac{d_{\min}}{r_a} + \frac{p^*}{P_k^*}$>=1: Chấp nhận x* là tâm cụm thứ (k+1); Tính lại mật độ cho các điểm còn lại theo công thức (.2); Và sang bước (k+1).

  + Ngược lại: Thiết lập P* = 0 và lặp lại bước k.

Nhận xét:

Các thuật toán phân cụm trước đó như k-means, c-means, k-medoids yêu cầu phải xác định trước số lượng cụm và đưa ra các tâm cụm khởi tạo ban đầu. Kết quả phân cụm phụ thuộc vào các tâm cụm ban đầu này. Việc tìm ra các tâm cụm khởi tạo để thu được kết quả tốt là vấn đề khó. Giải thuật mountain và phân cụm trừ đạ giải quyết được vấn đề này. Tự phân cụm và xác định tâm cụm dựa vào cấu trúc của tập dữ liệu.

So với giải thuật mountain đưa ra một không gian lưới và đánh giá khả năng trở thành tâm cụm của các điểm lưới dựa vào khoảng cách tới các điểm dữ liệu thực; Còn thuật toán phân cụm trừ đánh giá khả năng trở thành tâm cụm của các điểm dữ liệu thực dựa vào mật độ điểm lân cận.

Tuy nhiên, thuật toán FC phải thiết lập 4 tham số đầu vào: r_a, η (hay r_b), $\bar{\varepsilon}$ và $\underline{\varepsilon}$. Kết quả phân cụm phụ thuộc vào nhiều lựa chọn các tham số ban đầu và việc tìm các tham số để thuật toán chó kết quả tốt nhất đó là vấn đề khó khăn. Để khắc phục hạn chế này, một hướng cải tiến của thuật toán này là thuật toán phân cụm trừ mờ.

IV. THUẬT TOÁN PHÂN CỤM TRỪ MỜ

Để điều khiển kết quả phân cụm sao cho không phụ thuộc vào các tham số ban đầu, đưa tham số mờ m vào hàm tính mật độ cho các điểm dữ liệu như sau:

$$P_i = \sum_{j=1}^{n} e^{-\alpha \|x_i - x_j\|^{\frac{2}{m-1}}} \qquad (4)$$

Nếu x_k^* là vị trí tâm cụm thứ k, có mật độ là P_k^* thì mật độ cho các điểm còn lại tính theo công thức:

$$P_i = P_i - P_k^* e^{-\beta \|x_i - x_k\|^{\frac{2}{m-1}}} \quad (5)$$

Khi đó việc lựa chọn giá trị của tham số m sẽ ảnh hưởng rất lớn đến kết quả phân cụm. Nếu m càng lớn thì số lượng cụm tạo thành càng nhiều và ngược lại. Như vậy việc điều chỉnh giá trị tham số m cũng có thể thu được kết quả là tương đối tốt mà không phụ thuộc vào lựa chọn 4 tham số ban đầu.

V. ỨNG DỤNG CỦA PHÂN CỤM TRỪ VÀO PHÂN ĐOẠN ẢNH MÀU

Theo Chiu, các tham số ban đầu thường được chọn là: $r_a = 0.25, \eta = 1.5, \bar{\varepsilon} = 0.5, \underline{\varepsilon} = 0.15$. Tuy nhiê n, việc lựa chọn bán kín ra tuỳ thuộc vào tập dữ liệu. Trong ứng dụng này, tập dữ liệu ban đầu là tập các điểm ảnh có mức xám từ 0 đến 255. Do đó, ra là một giá trị trong khoảng 0-255, cụ thể ta chọn r_a = 15. Các tham số khác sử dụng là: $\eta = 1.5, \bar{\varepsilon} = 0.5, \underline{\varepsilon} = 0.15$. Kết quả như sau:

Ảnh ban đầu

Ảnh kết quả

được chia thành 20 cụm với các tâm cụm lần lượt và theo thứ tự của thuật toán là: 149, 175, 194, 162, 136, 216, 107, 28, 235, 7, 53, 75, 92, 121, 206, 40, 65, 17, 186, 227. Hình ảnh kết quả cho thấy đã có sự phâ n chia rõ ràng các vùng. các ảnh trong một vùng sẽ nhận mức xám của tâm cụm tương ứng. Như vậy ảnh kết quả có số mức xám thực bằng với số lượng cụm đã thu.

VI. KẾT LUẬN

Trên đây trình bày về thuật toán phân cụm trừ và ứng dụng vào việc phân đoạn ảnh màu. Để khắc phục hạn chế của thuật toán phân cụm trừ này, cũng đã đưa ra một cải tiến đó là thuật toán phân cụm trừ mờ.

TÀI LIỆU THAM KHẢO

1. Ngô Thành Long, Phạm Huy Bình, Phương pháp phân cụm mờ trừ loại hai khoảng, kỷ yếu hội nghị toàn quốc về điều khiển và tự động hoá – VCCA, 2011.

2. Trần Thị Yến và Bùi Đức Việt, Phương pháp phân cụm dữ liệu trừ và ứng dụng, tạp chí khoa học và công nghệ.

3. S. L. Chiu, Fuzzy Model Identification Based on Cluster Estimation, Journal on Intelligent uzzySystems, vol. 2, pp.267-278, 1994.

4. S. L. Chiu, Extracting Fuzzy Rules from Data for Function Approximation and PatternClassification, Fuzzy Information Engineering:a Guide Tour of Applications, pp.149-162.Wiley, New York, 1997.

5. Demirli, K., S. X. Cheng, P. Muthukumaran, Subtractive Clustering Based Modeling of obSequencing with Parametric Search, Fuzzy Setsand Systems. 137: 235-270, 2003.