

MỘT THUẬT TOÁN PHÁT HIỆN VÙNG VÀ ỨNG DỤNG CỦA NÓ TRONG QUÁ TRÌNH VÉC TƠ HÓA TỰ ĐỘNG

ĐỖ NĂNG TOÀN

Abstract. In this paper, we present an algorithm based on using the contours to detect region and its implementation in automatic vectorizing process. In general, regions are solid objects without holes inside and their sizes are bigger than the given threshold θ . The algorithm is used to select vectorizing method for each object in automatic vectorizing process.

1. GIỚI THIỆU

Việc chuyển đổi từ các bản đồ giấy sang bản đồ trên các máy tính nhờ các kỹ thuật véctơ hóa (tự động và bán tự động) trong những năm gần đây rất được quan tâm; đã có nhiều hệ thống dạng này như R2V, VPmaxNT, ImageIn, MapScan v.v... Có nhiều phương pháp véctơ hóa nhưng đối với bản đồ hay bản vẽ kỹ thuật có hai phương pháp chính:

- Phương pháp véctơ hóa theo xương thường được tiến hành dựa trên kỹ thuật làm mảnh, đối tượng được bóc dần lớp biên cho đến khi bị thu mảnh lại thành một đường duy nhất có độ dày bằng một pixel, sau đó liên kết các điểm xương với nhau thành các điểm kề nhau. Phương pháp này được áp dụng cho các đối tượng là các đoạn thẳng, đường tròn, cung tròn như đường ranh giới, đường bình độ... nhưng không thích hợp cho các đối tượng như ao, hồ...

- Phương pháp véctơ hóa theo đường biên được tiến hành dựa trên kỹ thuật dò biên, đối tượng sẽ được véctơ hóa theo các đường viền. Phương pháp này rất thích hợp đối với các đối tượng là ao, hồ...

Thực tế, bản đồ thường chứa cả hai loại đối tượng nêu trên, do đó nảy sinh vấn đề phân loại các đối tượng để lựa chọn phương pháp véctơ hóa thích hợp. Để thực hiện việc này có thể tiến hành một cách bán tự động nhờ việc chọn các vùng và chỉ định chế độ véctơ hóa thích hợp với vùng ấy. Việc này thường mất công và khó khăn do các đối tượng ở gần nhau. Bài báo này chỉ ra một thuật toán phát hiện vùng một cách tự động nhờ sử dụng chu tuyến và qua đó nêu ra một thuật toán véctơ hóa tự động kết hợp cả hai chế độ véctơ hóa (theo tâm và theo đường biên). Bài báo cũng đưa ra kết quả thực nghiệm để đánh giá tính hữu hiệu của thuật toán.

2. MỘT SỐ KHÁI NIỆM CƠ BẢN

• Ảnh và điểm ảnh

Ảnh được biểu diễn bằng một mảng số thực 2 chiều (a_{ij}) , kích thước $(m \times n)$, trong đó mỗi phần tử a_{ij} ; $i = 1, \dots, m$; $j = 1, \dots, n$ biểu thị mức xám của ảnh tại vị trí (i, j) tương ứng.

Một ảnh được gọi là nhị phân nếu các giá trị a_{ij} của nó chỉ nhận giá trị 0 hoặc 1.

Một ảnh bất kỳ có thể đưa về dạng nhị phân bằng phép cắt ngưỡng. Ta ký hiệu FF là tập điểm 1 (điểm ảnh) và \overline{FF} là tập các điểm 0 (điểm nền).

• Các điểm 4 và 8-láng giềng

Giả sử (i, j) là một điểm ảnh, các điểm 4-láng giềng là các điểm trực tiếp bên trên, dưới, trái, phải của điểm (i, j) :

$$N_4 = \{(i-1, j), (i+1, j), (i, j-1), (i, j+1)\},$$

và những điểm 8-láng giềng (hình 1) gồm:

$$N_8 = N_4 \cup \{(i-1, j-1), (i+1, j-1), (i-1, j+1), (i+1, j+1)\}.$$

Ví dụ trong hình 1 các điểm 0, 2, 4, 6 là các 4-láng giềng của điểm P , còn các điểm 0, 1, 2, 3, 4, 5, 6, 7 là các 8-láng giềng của P .

• Đối tượng ảnh

Hai điểm $P_1, P_2 \in E$, $E \subseteq FF$ hoặc \overline{FF} được gọi là 8-liên thông (hay 4-liên thông) trong E nếu tồn tại tập các điểm được gọi là "đường đi" $(i_0, j_0), \dots, (i_n, j_n)$ sao cho $(i_0, j_0) = P_1$, $(i_n, j_n) = P_2$, $(i_r, j_r) \in E$ và (i_r, j_r) là 8-láng giềng [hay 4-láng giềng] của (i_{r-1}, j_{r-1}) với $r = 1, 2, \dots, n$.

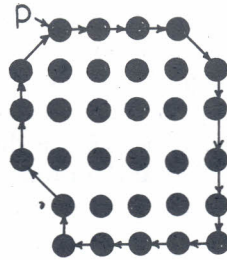
Quan hệ "k-liên thông trong E ", $k = 4, 8$ là một quan hệ phản xạ đối xứng và bắc cầu bởi vậy là một quan hệ tương đương. Mỗi lớp tương đương của nó được gọi là một thành phần k-liên thông của ảnh. Về sau ta sẽ gọi mỗi thành phần k-liên thông của ảnh là một đối tượng ảnh.

• Chu tuyến

Chu tuyến của một đối tượng ảnh là dãy các điểm của đối tượng ảnh p_0, p_1, \dots, p_N sao cho p_{i-1} là 8-láng giềng của p_i , $i = 1, \dots, N$; ngoài ra, $\forall i = 1, \dots, N$, $\exists q$ không thuộc đối tượng ảnh và q là 4-láng giềng của p_i và $p_0 \equiv p_N$. Hình 2 biểu diễn chu tuyến của ảnh, p là điểm khởi đầu chu tuyến.

3	2	1
4	P	0
5	6	7

Hình 1. Ma trận 8-láng giềng kề nhau



Hình 2. Ví dụ về chu tuyến của ảnh

• Chu tuyến láng giềng

Hai chu tuyến $C = \langle P_1 P_2 \dots P_n \rangle$ và $C_1 = \langle Q_1 Q_2 \dots Q_m \rangle$ được gọi là láng giềng của nhau nếu và chỉ nếu:

1. $P_i \neq Q_j \forall i, j$.
2. $\forall i \exists j$ sao cho P_i và Q_j là 8-láng giềng của nhau.
3. Các điểm P_i là ảnh thì Q_j là nền và ngược lại.

• Độ dài của chu tuyến

Độ dài của chu tuyến với các hướng như hình 2 được tính theo công thức sau:

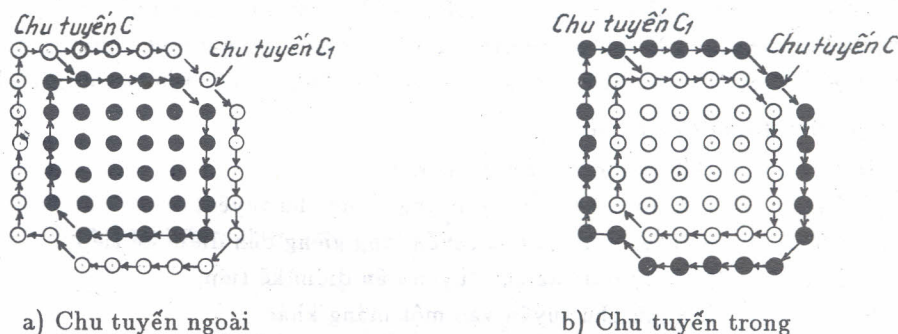
$$\text{PERIMETER} = \text{EVENPERIM} + \text{ODDPERIM} * \sqrt{2},$$

trong đó EVENPERIM là số lượng véc tơ theo hướng chẵn, còn ODDPERIM là số lượng véc tơ theo hướng lẻ.

• Chu tuyến ngoài, chu tuyến trong

C là chu tuyến của đối tượng ảnh $E \subseteq FF$, C_1 là chu tuyến láng giềng của chu tuyến C . Khi đó:

Nếu độ dài của chu tuyến C_1 lớn hơn độ dài chu tuyến C thì C_1 được gọi là chu tuyến ngoài. Nếu độ dài của chu tuyến C_1 nhỏ hơn độ dài chu tuyến C thì C_1 được gọi là chu tuyến trong.



Hình 3. Chu tuyến trong, ngoài tuyến ngoài

• Vùng

C là chu tuyến của đối tượng ảnh $E \subseteq EF$, θ là số thực > 0 . Khi đó E được gọi là vùng nếu thỏa mãn các điều kiện sau:

- Không có lỗ hổng [5] bên trong.
- Tồn tại một hình vuông có cạnh lớn hơn θ và nằm trọn trong E .

Trong trường hợp này C được gọi là chu tuyến xác định vùng.

3. PHÁT HIỆN VÙNG TRONG QUÁ TRÌNH VÉC TƠ HÓA TỰ ĐỘNG

Phần này đề cập tới phương pháp phát hiện vùng dựa vào chu tuyến, phương pháp này cho phép chọn kiểu véc tơ thích hợp cho từng đối tượng ngay trong quá trình véc tơ hóa tự động có sử dụng thuật toán làm mảnh theo chu tuyến.

3.1. Phát hiện vùng dựa vào chu tuyến

Một vấn đề quan trọng trong việc lựa chọn phương pháp véc tơ thích hợp cho từng đối tượng là phát hiện ra đặc tính (là vùng hay không) của chúng một cách tự động, trên cơ sở đó có thể áp dụng phương pháp véc tơ hóa thích hợp: véc tơ hóa theo biên và véc tơ hóa theo xương. Mệnh đề sau giúp cho chúng ta giảm bớt kiểm tra một đối tượng có phải vùng hay không.

Mệnh đề 1. Nếu C là chu tuyến xác định vùng khi đó C là chu tuyến ngoài.

Chứng minh. Gọi $C = \langle P_1 P_2 \dots P_n \rangle$, $C_1 = \langle Q_1 Q_2 \dots Q_m \rangle$ là chu tuyến láng giềng của C . Ta phải chứng minh độ dài của chu tuyến C_1 ($\text{Len}(C_1)$) lớn hơn độ dài của chu tuyến C ($\text{Len}(C)$).

Thật vậy, giả sử $\text{Len}(C) > \text{Len}(C_1)$ theo mệnh đề phát hiện lỗ hổng [5] ta có C là chu tuyến xác định lỗ hổng.

Mặt khác, theo giả thiết C là chu tuyến xác định vùng, nên theo định nghĩa thì không tồn tại lỗ hổng bên trong C .

Từ đó suy ra mâu thuẫn, vậy $\text{Len}(C) < \text{Len}(C_1)$.

Do đó, theo định nghĩa C là chu tuyến ngoài. □

Phát hiện vùng

Thuật toán CHECK_REGION sau đây kiểm tra đối tượng tương ứng với một chu tuyến cho trước có phải là vùng hay không, nếu đúng trả lại giá trị TRUE, nếu sai trả lại giá trị FALSE.

Các biến sử dụng trong thuật toán

- 1 $StartPT$, $StartDir$ điểm và hướng khởi đầu của chu tuyến (hình 2),
- 2 $NextPT$, $NextDir$ điểm và hướng kế tiếp theo chiều xét duyệt chu tuyến (hình 2),

- 3 $nWhite, nBack$ độ dài của chu tuyến và chu tuyến láng giềng,
 4 $nWidth, nHeight$ độ dài và độ cao của đối tượng tại điểm hiện thời,
 5 $HozPT, VerPT$ các điểm cùng tung độ và hoành độ với điểm đang xét của chu tuyến.

Các hàm sử dụng trong thuật toán

- 1 *Init* thiết lập các tham số ban đầu,
 2 *FindNext* tìm điểm kế tiếp và hướng trong chu tuyến,
 3 *LenWhite* tính độ dài của chu tuyến láng giềng đến điểm kế tiếp,
 LenBack tính độ dài của chu tuyến đến điểm kế tiếp,
 4 *PutDest* lưu giữ chu tuyến vào một mảng khác,
 5 *CalculateHoz* tính chiều ngang và tìm điểm biên tương ứng,
 CalculateVer tính chiều dọc và tìm điểm biên tương ứng,
 6 *CheckDest* kiểm tra xem điểm đã cho có nằm trong chu tuyến không (có nằm trong mảng đã đánh dấu chưa),
 7 *Check θ* kiểm tra trong hình vuông có cạnh là θ xem có điểm nền (lỗ hổng) nào không.

Thuật toán

```
//Kiểm tra chu tuyến
Ini;
Repeat
    FindNext(&NextPT,&NextDir);
    nWhite := nWhite + LenWhite(NextPT);
    nBlack := nBlack + LenBlack(NextPT);
    PutDest(NextPT);
Until NextPT==StartPT And NextDir==StartDir;
If nWhite ≤ nBlack Then
    Return FALSE;
//Kiểm tra lỗ thủng và ngưỡng  $\theta$ 
Repeat
    FindNext(&NextPt,&NextDir);
    CalculateHoz(NextPT,&nWidth,&HozPT);
    CalculateVer(NextPT,&nHeight,&VerPT);
    If not CheckDest(HozPT) or not CheckDest(VerPT) Then
        Return FALSE;
    If nWidth ≥  $\theta$  And nHeight ≥  $\theta$  And not fLag $\theta$  Then
        If Check $\theta$ (NextPT) Then
            fLag $\theta$ :=TRUE;
Until NextPT==StartPT And NextDir==StartDir;
//Kết thúc
Return fLag $\theta$ ;
```

Mệnh đề 2. Thuật toán CHECK_REGION dừng và cho kết quả đúng.

Chứng minh. Chúng ta biết số điểm của chu tuyến và đối tượng xác định bởi chu tuyến là hữu hạn nên các bước kiểm tra chu tuyến và kiểm tra lỗ hổng và ngưỡng đều dừng do đó thuật toán CHECK_REGION cũng dừng.

Từ bước kiểm tra lỗ hổng cho thấy không tồn tại lỗ hổng bên trong đối tượng. Thật vậy, giả sử tồn tại một lỗ hổng bên trong đối tượng, khi đó nếu xét theo chiều ngang hoặc chiều đứng do

đối tượng là hữu hạn nên tồn tại một điểm thuộc chu tuyến xác định lỗ hổng và một điểm của chu tuyến đang xét có cùng tung độ hoặc hoành độ. Trong thuật toán điểm thuộc chu tuyến xác định lỗ hổng này chính là điểm HozPT hoặc VerPT. Nhưng theo thuật toán thì tất cả các điểm HozPT, VerPT này đều phải thuộc chu tuyến đang xét. Từ đó suy ra mâu thuẫn, vậy không tồn tại lỗ hổng bên trong đối tượng. (*)

Hơn nữa, từ bước kiểm tra ngưỡng cho thấy đối tượng tìm được sẽ tồn tại một hình vuông có cạnh θ nằm trọn trong chu tuyến. (**)

Từ (*) và (**) theo định nghĩa ta có chu tuyến xác định vùng. \square

3.2. Thuật toán véc tơ hóa kết hợp (theo tâm và theo biên)

Trong thuật toán véc tơ hóa sử dụng phương pháp làm mảnh theo chu tuyến (xem [1, 3, 4, 6]) có bước tìm duyệt các chu tuyến. Để quyết định việc lựa chọn chế độ véc tơ cho thích hợp đối với đối tượng (theo tâm hay theo đường biên), thuật toán sử dụng đồng thời các chu tuyến kết quả tìm được trong bước này và sử dụng thuật toán CHECK_REGION trong mục 3.1 để xác định đối tượng có phải là vùng hay không.

Thuật toán véc tơ hóa tự động kết hợp dựa theo chu tuyến gồm các bước sau:

Bước 1 [Duyệt tìm chu tuyến]:

Việc duyệt tìm được tiến hành từ trái sang phải, từ trên xuống dưới theo ma trận các điểm ảnh.

Nếu không thấy chu tuyến nào nữa thì nhảy qua thực hiện bước 2, ngược lại sử dụng thuật toán CHECK_REGION để kiểm tra xem chu tuyến vừa tìm được có phải là chu tuyến xác định vùng hay không? Nếu là chu tuyến xác định vùng thì tính chất vùng của chu tuyến sẽ được đánh dấu.

Chu tuyến kết quả vừa tìm thấy cùng với thuộc tính của nó sẽ được lưu vào một mảng.

Tiếp tục thực hiện bước 1.

Bước 2 [Làm mảnh]:

Với mỗi đối tượng là chu tuyến tìm được trong mảng lưu trữ có thuộc tính không phải là thuộc tính xác định vùng, ta tiến hành bước làm mảnh. Việc làm mảnh bao gồm nhiều lần gập, trong mỗi lần lặp tất cả các điểm của đối tượng sẽ được kiểm tra nếu như chúng thỏa mãn điều kiện xóa thì sẽ bị xóa đi. Quá trình được lặp lại cho đến khi không còn điểm biên nào được xóa. Đối tượng được bóc dần lớp biên cho đến khi bị thu mảnh lại thành một đường duy nhất có độ dày bằng một pixel.

Nếu không còn chu tuyến nào nữa thì nhảy qua thực hiện bước 3, ngược lại tiếp tục thực hiện bước 2.

Bước 3 [Véc tơ hóa]:

Với các đối tượng đã được làm mảnh sẽ được tiến hành véc tơ hóa bởi việc dò theo xương, còn đối với các đối tượng được xác định bởi chu tuyến có tính chất vùng sẽ được tiến hành véc tơ hóa bởi việc dò biên. Kết quả thu được là một dãy điểm liên tiếp, để giảm việc lưu trữ chúng sẽ được loại bỏ bớt nhờ các thuật toán đơn giản hóa (Douglas-Peucker, Angler, Width-Band...).

Nếu không còn chu tuyến nào nữa thì dừng, ngược lại tiếp tục thực hiện bước 3.

4. THỰC NGHIỆM

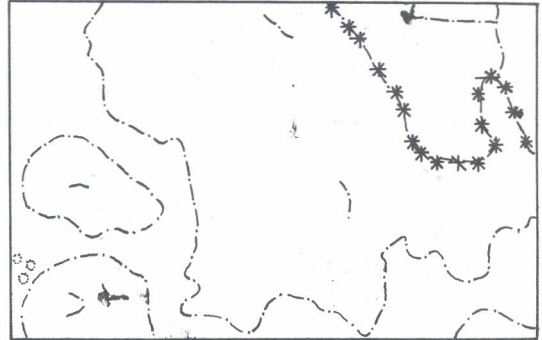
Chúng tôi đã cài đặt các phương pháp này trong hệ mềm MAPSCAN. Phương pháp này tỏ ra rất hữu hiệu đối với các bản đồ bình độ, thủy văn, đường biên hành chính, đường giao thông, v.v...

Hình 4a là ảnh gốc trước khi tiến hành véc tơ hóa. Hình 4b là ảnh véc tơ kết quả thực hiện véc tơ hóa theo xương, các đối tượng là những vùng như ao, hồ... bị biến thành điểm hoặc đoạn thẳng. Hình 4c là ảnh véc tơ kết quả thực hiện véc tơ hóa theo biên, các đối tượng đều được véc tơ hóa theo đường viền bao gồm cả các đối tượng là đường ranh giới, đường bình độ... Hình 4d là ảnh

véc tơ kết quả thực hiện theo thuật toán véc tơ hóa kết hợp nêu ở phần 3, các đối tượng là đường ranh giới, đường bình độ... thì được véc tơ hóa theo xương, còn đối tượng như ao, hồ... thì được véc tơ hóa theo đường biên.



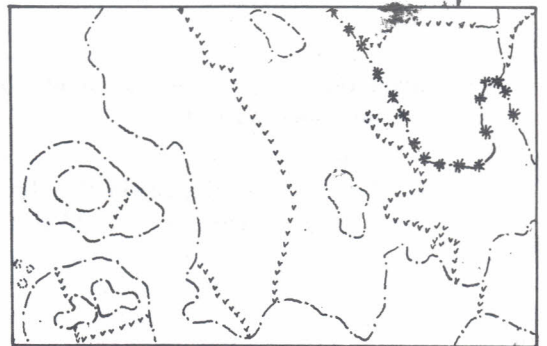
a) Ảnh gốc trước khi tiến hành véc tơ hóa



b) Véc tơ hóa theo xương



c) Véc tơ hóa theo biên



d) Véc tơ hóa kết hợp

Hình 4. Các chế độ véc tơ hóa

5. KẾT LUẬN

Trong bài báo này chúng tôi đã đưa ra một định nghĩa của vùng, tiếp đó là một thuật toán phát hiện vùng ngay trong quá trình xét duyệt chu tuyến. Thuật toán này có thể được sử dụng đồng thời với quá trình véc tơ hóa tự động có sử dụng thuật toán làm mảnh theo chu tuyến để điều chỉnh phương pháp véc tơ hóa. Qua thực nghiệm chúng tôi thấy phương pháp phát hiện vùng đảm bảo việc véc tơ hóa kết hợp cả hai phương pháp theo xương và theo biên là nhanh và chính xác.

Lời cảm ơn

Tôi xin chân thành cảm ơn GS TS Bạch Hưng Khang đã tận tình giúp đỡ tôi trong quá trình nghiên cứu và hoàn thiện thuật toán này. Tôi cũng bày tỏ lòng biết ơn đến PTS Ngô Quốc Tạo, PTS Lương Chi Mai đã đóng góp những ý kiến quý báu giúp tôi hoàn thành bài báo này. Cuối cùng tôi xin chân thành cảm ơn các đồng nghiệp phòng Nhận dạng và Công nghệ tri thức đã tạo điều kiện thuận lợi cho tôi trong cả việc nghiên cứu lẫn việc cài đặt thuật toán.

TÀI LIỆU THAM KHẢO

- [1] Bạch Hưng Khang, Lương Chi Mai, Ngô Quốc Tạo, Đỗ Năng Toàn, et al., An Examination of Techniques for Raster-to-Vector Process and Its Implementation-MapScan Package Software, *International Symposium AMPST96*, University of Bradford, UK, March 1996.
- [2] P. A. Burrough, Principles of Geographical Information System for Land Resources Assessment, 1987, 30-31.
- [3] Ngô Quốc Tạo, Đặng Ngọc Đức, Thuật toán làm mảnh tuần tự mới, *Tuyển tập Báo cáo hội nghị khoa học Viện Công nghệ thông tin*, Hà Nội, 1996.
- [4] Nguyễn Ngọc Kỳ, *Biểu diễn và đồng nhất tự động ảnh đường nét*, Luận án Phó tiến sĩ Toán - Lý, Hà Nội, 1992.
- [5] Đỗ Năng Toàn, Ứng dụng chu tuyến vào việc loại bỏ đối tượng nhỏ trong quá trình véc tơ hóa tự động, *Tạp chí Tin học và Điều khiển học* **15** (2) (1999) 40-46.
- [6] P. S. P. Wang and Y. Y. Zhang, A Fast and Flexible Thinning Algorithms, *IEEE Transactions on Computer* **38** (1989) 741-745.

Nhận bài ngày 14 - 4 - 1999

Nhận lại sau khi sửa ngày 5 - 12 - 1999

Phòng Nhận dạng và Công nghệ tri thức
Viện Công nghệ thông tin