

MỤC LỤC

LỜI CẢM ƠN	3
LỜI MỞ ĐẦU	3
Chương 1: KHÁI QUÁT VỀ XỬ LÝ ẢNH VÀ XƯƠNG ẢNH	5
1.1. Khái quát về xử lý ảnh	5
1.1.1. Xử lý ảnh, sơ đồ của hệ thống xử lý ảnh	5
1.1.2. Một số vấn đề cơ bản của xử lý ảnh	6
1.2. Xương và phát hiện xương trong xử lý ảnh	10
Chương 2: KỸ THUẬT PHÁT HIỆN XƯƠNG NHỜ PHÉP TOÁN HÌNH THÁI ...	13
2.1. Các phép toán hình thái và các tính chất	13
2.1.1. Khái niệm cơ bản	13
2.1.1.1. Phép giãn nhị phân (Dilation)	17
2.1.1.2. Phép co nhị phân (Erosion).....	19
2.1.2. Một số tính chất của phép toán hình thái	23
2.1.2.1. Các mệnh đề.....	23
2.1.2.2. Định lý	23
2.1.2.3. Hệ quả	24
2.2. Kỹ thuật tìm xương nhờ phép toán hình thái	25
2.2.1. Trích biên (Boundary Extraction).....	25
2.2.2. Làm đầy (Region Filling).....	26
2.2.3. Tách các thành phần liên thông (Extraction of Connected Components)	27
2.2.4. Bao Lồi (Convex Hull)	29
2.2.5. Làm mảnh (Thinning)	31
2.2.6. Tìm khung xương (Skeletonization).....	33
Chương 3: CHƯƠNG TRÌNH THỬ NGHIỆM	35
3.1. Bài toán	35

3.2. Phân tích và giới thiệu	36
3.3. Một số kết quả của chương trình	37
PHẦN KẾT LUẬN	42
TÀI LIỆU THAM KHẢO	43

LỜI CẢM ƠN!

Trước hết em xin bày tỏ lòng biết ơn sâu sắc nhất tới PGS. TS. ĐỖ NĂNG TOÀN đã tận tình giúp đỡ em rất nhiều trong suốt quá trình tìm hiểu nghiên cứu và hoàn thành báo cáo tốt nghiệp.

Em xin chân thành cảm ơn các thầy cô trong bộ môn tin học – trường DHDL Hải Phòng cũng như các thầy cô trong trường đã trang bị cho em những kiến thức cơ bản cần thiết để em có thể hoàn thành báo cáo.

Xin gửi lời cảm ơn đến bạn bè những người luôn bên em đã động viên và tạo điều kiện thuận lợi cho em, tận tình giúp đỡ chỉ bảo em những gì em còn thiếu sót trong quá trình làm báo cáo tốt nghiệp.

Cuối cùng em xin bày tỏ lòng biết ơn sâu sắc tới những người thân trong gia đình đã giành cho em sự quan tâm đặc biệt và luôn động viên em.

Vì thời gian có hạn, trình độ hiểu biết của bản thân còn nhiều hạn chế. Cho nên trong đề án không tránh khỏi những thiếu sót, em rất mong nhận được sự đóng góp ý kiến của tất cả các thầy cô giáo cũng như các bạn bè để đề án của em được hoàn thiện hơn.

Em xin chân thành cảm ơn!

Hải phòng, ngày... tháng...năm 2012

Sinh viên thực hiện

Bùi Duy Mạnh

Lời mở đầu

Trong thực tế hình dạng thường được chú trọng hơn kích thước và con người nhận ra các đối tượng xung quanh cũng chủ yếu thông qua hình dạng. Chính vì vậy, biểu diễn hình dạng là một vấn đề quan trọng và không thể thiếu trong quá trình nhận dạng đối tượng. Trong xử lý ảnh số ta sử dụng các phép toán hình thái để giải quyết vấn đề này.

Các phép toán hình thái không những quan trọng trong lĩnh vực toán học mà còn rất quan trọng trong lĩnh vực Công Nghệ Thông Tin nói chung và xử lý ảnh số nói riêng: giúp ta giải quyết các bài toán như tìm biên ảnh, lọc nhiễu trên ảnh, cải thiện chất lượng hình ảnh, nối các nét đứt trong văn bản đã xuống cấp và phát hiện đối tượng chuyển động và nhiều các ứng dụng các nữa.

Chính vì vậy em đã lựa chọn đề tài “Tìm hiểu bài toán phát hiện xương nhờ phép toán hình thái”.

Mục tiêu của đề án là phát hiện xương nhờ phép toán hình thái vào xử lý các đối tượng trong hình ảnh.

Ban đầu em đã tìm đọc các tài liệu và sách báo có nội dung liên quan đến các khái niệm trong đề tài, sau đó đưa ra phương hướng, và rút ra kết luận cho riêng mình để thực hiện đề tài. Báo cáo đề án gồm 44 trang, được chia làm 3 chương:

Chương I. Khái quát về xử lý ảnh và xương ảnh: Chương này gồm có các khái niệm ban đầu về xử lý ảnh số, sơ đồ của hệ thống xử lý ảnh, một số vấn đề cơ bản của xử lý ảnh, xương và phát hiện xương trong xử lý ảnh

Chương II. Kỹ thuật phát hiện xương nhờ phép toán hình thái: Chương này gồm có các khái niệm về phép toán hình thái, từ cơ bản đến những khái niệm nâng cao và ứng dụng trong thực tế. Toàn bộ chương tập trung làm rõ các khái niệm về: Phép co nhị phân, phép giãn nhị phân, một số tính chất của phép toán hình thái. Các kỹ thuật tìm xương nhờ phép toán hình thái phép trích biên, phép làm đầy, phép tách các thành phần liên thông, phép bao lồi, phép làm mảnh và tìm xương.

Chương III. Chương trình thử nghiệm: Trên cơ sở lý thuyết đã tìm hiểu em xây dựng chương trình xử lý ảnh sử dụng phép toán hình thái: tìm xương các đối tượng trên ảnh nhị phân.

Kết Luận...

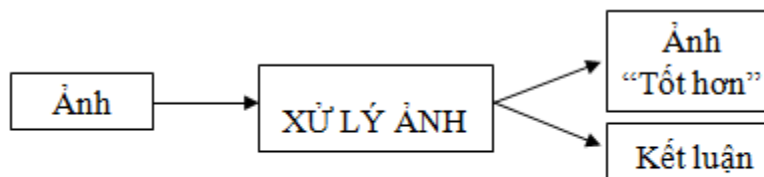
Chương 1: KHÁI QUÁT VỀ XỬ LÝ ẢNH VÀ XƯƠNG ẢNH

1.1. Khái quát về xử lý ảnh

1.1.1. Xử lý ảnh, sơ đồ của hệ thống xử lý ảnh

Con người thu nhận thông tin qua các giác quan, trong đó thị giác đóng vai trò quan trọng nhất. Những năm trở lại đây với sự phát triển của phần cứng máy tính, xử lý ảnh và đồ họa đã phát triển một cách mạnh mẽ và có nhiều ứng dụng trong cuộc sống. Xử lý ảnh và đồ họa đóng một vai trò quan trọng trong tương tác người máy.

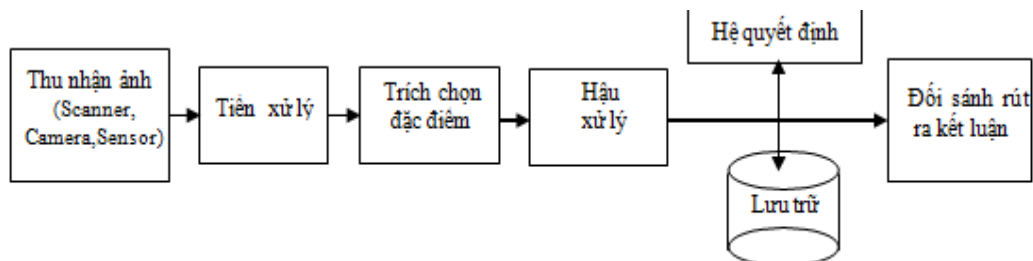
Quá trình xử lý ảnh được xem như là quá trình thao tác ảnh đầu vào nhằm cho ra kết quả mong muốn. Kết quả đầu ra của một quá trình xử lý ảnh có thể là một ảnh “tốt hơn” hoặc một kết luận.



Hình 1.1. Quá trình xử lý ảnh

Ảnh có thể xem là tập hợp các điểm ảnh và mỗi điểm ảnh được xem như là đặc trưng cường độ sáng hay một dấu hiệu nào đó tại một vị trí nào đó của đối tượng trong không gian và nó có thể xem như một hàm n biến $P(c_1, c_2, \dots, c_n)$. Do đó, ảnh trong xử lý ảnh có thể xem như ảnh n chiều.

Sơ đồ tổng quát của một hệ thống xử lý ảnh:



Hình 1.2. Các bước cơ bản trong một hệ thống xử lý ảnh

1.1.2. Một số vấn đề cơ bản của xử lý ảnh

1.1.2.1 Một số khái niệm cơ bản

* Ảnh và điểm ảnh:

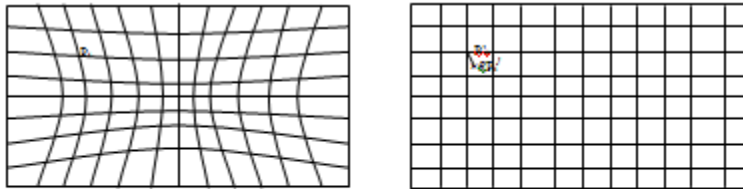
Điểm ảnh được xem như là dấu hiệu hay cường độ sáng tại 1 tọa độ trong không gian của đối tượng và ảnh được xem như là 1 tập hợp các điểm ảnh.

* Mức xám, màu

Là số các giá trị có thể có của các điểm ảnh của ảnh

1.1.2.2 Nắn chỉnh biến dạng

Ảnh thu nhận thường bị biến dạng do các thiết bị quang học và điện tử.



Ảnh thu nhận

Ảnh mong muốn

Hình 1.3. Ảnh thu nhận và ảnh mong muốn

Để khắc phục người ta sử dụng các phép chiếu, các phép chiếu thường được xây dựng trên tập các điểm điều khiển.

Giả sử (P_i, P_i') $i = 1, n$ có n các tập điều khiển

Tìm hàm $f: P_i \rightarrow f(P_i)$ sao cho

$$\sum_{i=1}^n |f(P_i) - P_i'|^2 \rightarrow \min$$

Giả sử ảnh bị biến đổi chỉ bao gồm: Tịnh tiến, quay, tỷ lệ, biến dạng bậc nhất tuyến tính.

Khi đó hàm f có dạng:

$$f(x, y) = (a_1x + b_1y + c_1, a_2x + b_2y + c_2)$$

Ta có:

$$\phi = \sum_{i=1}^n (f(P_i) - P_i')^2 = \sum_{i=1}^n \left[(a_1 x_i + b_1 y_i + c_1 - x_i')^2 + (a_2 x_i + b_2 y_i + c_2 - y_i')^2 \right]$$

Để cho $\phi \rightarrow \min$

$$\begin{cases} \frac{\partial \phi}{\partial a_1} = 0 \\ \frac{\partial \phi}{\partial b_1} = 0 \\ \frac{\partial \phi}{\partial c_1} = 0 \end{cases} \Leftrightarrow \begin{cases} \sum_{i=1}^n a_1 x_i^2 + \sum_{i=1}^n b_1 x_i y_i + \sum_{i=1}^n c_1 x_i = \sum_{i=1}^n x_i x_i' \\ \sum_{i=1}^n a_1 x_i y_i + \sum_{i=1}^n b_1 y_i^2 + \sum_{i=1}^n c_1 y_i = \sum_{i=1}^n y_i x_i' \\ \sum_{i=1}^n a_1 x_i + \sum_{i=1}^n b_1 y_i + n c_1 = \sum_{i=1}^n x_i' \end{cases}$$

Giải hệ phương trình tuyến tính tìm được a_1, b_1, c_1

Tương tự tìm được a_2, b_2, c_2

\Rightarrow Xác định được hàm f

1.1.2.3 Khử nhiễu

Có 2 loại nhiễu cơ bản trong quá trình thu nhận ảnh

- Nhiễu hệ thống: là nhiễu có quy luật có thể khử bằng các phép biến đổi
- Nhiễu ngẫu nhiên: vết bản không rõ nguyên nhân \rightarrow khắc phục bằng các phép lọc

1.1.2.4 Chỉnh mức xám:

Nhằm khắc phục tính không đồng đều của hệ thống gây ra. Thông thường có 2 hướng tiếp cận:

- Giảm số mức xám: Thực hiện bằng cách nhóm các mức xám gần nhau thành một bó. Trường hợp chỉ có 2 mức xám thì chính là chuyển về ảnh đen trắng. Ứng dụng: In ảnh màu ra máy in đen trắng.
- Tăng số mức xám: Thực hiện nội suy ra các mức xám trung gian bằng kỹ thuật nội suy. Kỹ thuật này nhằm tăng cường độ mịn cho ảnh

1.1.2.5 Trích chọn đặc điểm

Các đặc điểm của đối tượng được trích chọn tùy theo mục đích nhận dạng trong quá trình xử lý ảnh. Có thể nêu ra một số đặc điểm của ảnh sau đây:

Đặc điểm không gian: Phân bố mức xám, phân bố xác suất, biên độ, điểm uốn v.v..

Đặc điểm biến đổi: Các đặc điểm loại này được trích chọn bằng việc thực hiện lọc vùng (zonal filtering). Các bộ vùng được gọi là “mặt nạ đặc điểm” (feature mask) thường là các khe hẹp với hình dạng khác nhau (chữ nhật, tam giác, cung tròn v.v..)

Đặc điểm biên và đường biên: Đặc trưng cho đường biên của đối tượng và do vậy rất hữu ích trong việc trích chọn các thuộc tính bất biến được dùng khi nhận dạng đối tượng. Các đặc điểm này có thể được trích chọn nhờ toán tử gradient, toán tử la bàn, toán tử Laplace, toán tử “chéo không” (zero crossing) v.v..

Việc trích chọn hiệu quả các đặc điểm giúp cho việc nhận dạng các đối tượng ảnh chính xác, với tốc độ tính toán cao và dung lượng nhớ lưu trữ giảm xuống.

1.1.2.6 Nhận dạng

Nhận dạng tự động (automatic recognition), mô tả đối tượng, phân loại và phân nhóm các mẫu là những vấn đề quan trọng trong thị giác máy, được ứng dụng trong nhiều ngành khoa học khác nhau. Tuy nhiên, một câu hỏi đặt ra là: mẫu (pattern) là gì? Watanabe, một trong những người đi đầu trong lĩnh vực này đã định nghĩa: “Ngược lại với hỗn loạn (chaos), mẫu là một thực thể (entity), được xác định một cách ang áng (vaguely defined) và có thể gán cho nó một tên gọi nào đó”. Ví dụ mẫu có thể là ảnh của vân tay, ảnh của một vật nào đó được chụp, một chữ viết, khuôn mặt người hoặc một ký đồ tín hiệu tiếng nói. Khi biết một mẫu nào đó, để nhận dạng hoặc phân loại mẫu đó có thể:

Hoặc phân loại có mẫu (supervised classification), chẳng hạn phân tích phân biệt (discriminant analysis), trong đó mẫu đầu vào được định danh như một thành phần của một lớp đã xác định.

Hoặc phân loại không có mẫu (unsupervised classification hay clustering) trong đó các mẫu được gán vào các lớp khác nhau dựa trên một tiêu chuẩn đồng dạng nào đó. Các lớp này cho đến thời điểm phân loại vẫn chưa biết hay chưa được định danh.

Hệ thống nhận dạng tự động bao gồm ba khâu tương ứng với ba giai đoạn chủ yếu

sau đây:

- 1°. Thu nhận dữ liệu và tiền xử lý.
- 2°. Biểu diễn dữ liệu.
- 3°. Nhận dạng, ra quyết định.

Bốn cách tiếp cận khác nhau trong lý thuyết nhận dạng là:

- 1°. Đối sánh mẫu dựa trên các đặc trưng được trích chọn.
- 2°. Phân loại thống kê.
- 3°. Đối sánh cấu trúc.
- 4°. Phân loại dựa trên mạng nơ-ron nhân tạo.

Trong các ứng dụng rõ ràng là không thể chỉ dùng có một cách tiếp cận đơn lẻ để phân loại “tối ưu” do vậy cần sử dụng cùng một lúc nhiều phương pháp và cách tiếp cận khác nhau. Do vậy, các phương thức phân loại tổ hợp hay được sử dụng khi nhận dạng và nay đã có những kết quả có triển vọng dựa trên thiết kế các hệ thống lai (hybrid system) bao gồm nhiều mô hình kết hợp.

Việc giải quyết bài toán nhận dạng trong những ứng dụng mới, nảy sinh trong cuộc sống không chỉ tạo ra những thách thức về thuật giải, mà còn đặt ra những yêu cầu về tốc độ tính toán. Đặc điểm chung của tất cả những ứng dụng đó là những đặc điểm đặc trưng cần thiết thường là nhiều, không thể do chuyên gia đề xuất, mà phải được trích chọn dựa trên các thủ tục phân tích dữ liệu.

1.1.2.7 Nén ảnh

Nhằm giảm thiểu không gian lưu trữ. Thường được tiến hành theo cả hai cách khuynh hướng là nén có bảo toàn và không bảo toàn thông tin. Nén không bảo toàn thì thường có khả năng nén cao hơn nhưng khả năng phục hồi thì kém hơn. Trên cơ sở hai khuynh hướng, có 4 cách tiếp cận cơ bản trong nén ảnh:

- Nén ảnh thống kê: Kỹ thuật nén này dựa vào việc thống kê tần xuất xuất hiện của giá trị các điểm ảnh, trên cơ sở đó mà có chiến lược mã hóa thích hợp. Một ví dụ điển hình cho kỹ thuật mã hóa này là *.TIF
- Nén ảnh không gian: Kỹ thuật này dựa vào vị trí không gian của các điểm ảnh để tiến hành mã hóa. Kỹ thuật lợi dụng sự giống nhau của các điểm ảnh trong các vùng gần

nhau. Ví dụ cho kỹ thuật này là mã nén *.PCX

- Nén ảnh sử dụng phép biến đổi: Đây là kỹ thuật tiếp cận theo hướng nén không bảo toàn và do vậy, kỹ thuật thường nén hiệu quả hơn. *.JPG chính là tiếp cận theo kỹ thuật nén này.

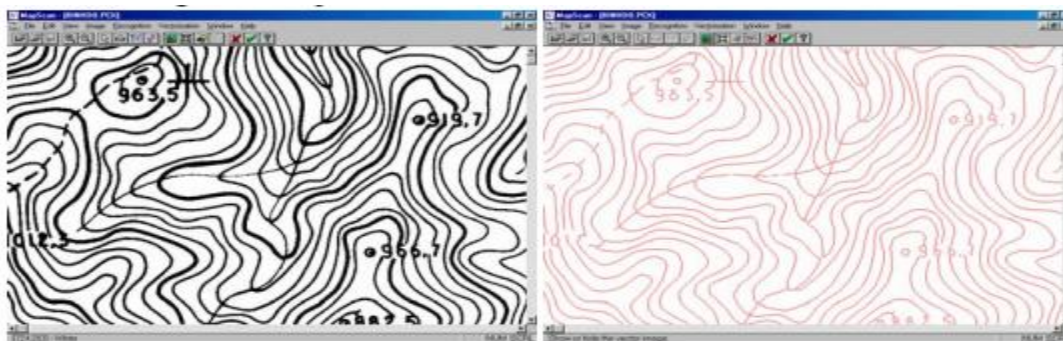
- Nén ảnh Fractal: Sử dụng tính chất Fractal của các đối tượng ảnh, thể hiện sự lặp lại của các chi tiết. Kỹ thuật nén sẽ tính toán để chỉ cần lưu trữ phần gốc ảnh và quy luật sinh ra ảnh theo nguyên lý Fractal

1.2. Xương và phát hiện xương trong xử lý ảnh

1.2.1. Giới Thiệu

Xương được coi như hình dạng cơ bản của một đối tượng, với số ít các điểm ảnh cơ bản. Ta có thể lấy được các thông tin về hình dạng nguyên bản của một đối tượng thông qua xương.

Một định nghĩa xúc tích về xương dựa trên tính continuum (tương tự như hiện tượng cháy đồng cỏ) được đưa ra bởi Blum (1976) như sau: Giả thiết rằng đối tượng là đồng nhất được phủ bởi cỏ khô và sau đó dựng lên một vòng biên lửa. Xương được định nghĩa như nơi gặp của các vết lửa và tại đó chúng được dập tắt.



a)Ảnh gốc

b)Ảnh xương

Hình 1.4. Ví dụ về ảnh và xương

Kỹ thuật tìm xương luôn là chủ đề nghiên cứu trong xử lý ảnh những năm gần đây. Mặc dù có những nỗ lực cho việc phát triển các thuật toán tìm xương, nhưng các phương pháp được đưa ra đều bị mất mát thông tin. Có thể chia thành hai loại thuật toán tìm xương cơ bản:

- Các thuật toán tìm xương dựa trên làm mảnh
- Các thuật toán tìm xương không dựa trên làm mảnh

1.2.2. Tìm xương dựa trên làm mảnh

Thuật toán làm mảnh ảnh số nhị phân là một trong các thuật toán quan trọng trong xử lý ảnh và nhận dạng. Xương chứa những thông tin bất biến về cấu trúc của ảnh, giúp cho quá trình nhận dạng hoặc vectơ hoá sau này.

Thuật toán làm mảnh là quá trình lặp duyệt và kiểm tra tất cả các điểm thuộc đối tượng. Trong mỗi lần lặp tất cả các điểm của đối tượng sẽ được kiểm tra: nếu như chúng thoả mãn điều kiện xoá nào đó tùy thuộc vào mỗi thuật toán thì nó sẽ bị xoá đi. Quá trình cứ lặp lại cho đến khi không còn điểm biên nào được xoá. Đối tượng được bóc dần lớp biên cho đến khi nào bị thu mảnh lại chỉ còn các điểm biên.

Các thuật toán làm mảnh được phân loại dựa trên phương pháp xử lý các điểm là thuật toán làm mảnh song song và thuật toán làm mảnh tuần tự.

Thuật toán làm mảnh song song, là thuật toán mà trong đó các điểm được xử lý theo phương pháp song song, tức là được xử lý cùng một lúc. Giá trị của mỗi điểm sau một lần lặp chỉ phụ thuộc vào giá trị của các láng giềng bên cạnh (thường là 8-láng giềng) mà giá trị của các điểm này đã được xác định trong lần lặp trước đó. Trong máy có nhiều bộ vi xử lý mỗi vi xử lý sẽ xử lý một vùng của đối tượng, nó có quyền đọc từ các điểm ở vùng khác nhưng chỉ được ghi trên vùng của nó xử lý.

Trong thuật toán làm mảnh tuần tự các điểm thuộc đối tượng sẽ được kiểm tra theo một thứ tự nào đó (chẳng hạn các điểm được xét từ trái qua phải, trên xuống dưới). Giá trị của điểm sau mỗi lần lặp không những phụ thuộc vào giá trị của các láng giềng bên cạnh mà còn phụ thuộc vào các điểm đã được xét trước đó trong chính lần lặp đang xét.

Chất lượng của thuật toán làm mảnh được đánh giá theo các tiêu chuẩn được liệt kê dưới đây nhưng không nhất thiết phải thoả mãn đồng thời tất cả các tiêu chuẩn.

- Bảo toàn tính liên thông của đối tượng và phần bù của đối tượng
- Sự tương hợp giữa xương và cấu trúc của ảnh đối tượng
- Bảo toàn các thành phần liên thông
- Bảo toàn các điểm cụt
- Xương chỉ gồm các điểm biên, càng mảnh càng tốt

- Bền vững đối với nhiễu
- Xương cho phép khôi phục ảnh ban đầu của đối tượng
- Xương thu được ở chính giữa đường nét của đối tượng được làm mảnh
- Xương nhận được bất biến với phép quay.

1.2.3. Tìm xương không dựa trên làm mảnh

Để tách được xương của đối tượng có thể sử dụng đường biên của đối tượng. Với điểm p bất kỳ trên đối tượng, ta bao nó bởi một đường biên. Nếu như có nhiều điểm biên có cùng khoảng cách ngắn nhất tới p thì p nằm trên trục trung vị. Tập tất cả các điểm như vậy lập thành trục trung vị hay xương của đối tượng. Việc xác định xương được tiến hành thông qua hai bước:

- Bước thứ nhất, tính khoảng cách từ mỗi điểm ảnh của đối tượng đến điểm biên gần nhất. Như vậy cần phải tính toán khoảng cách tới tất cả các điểm biên của ảnh.
- Bước thứ hai, khoảng cách ảnh đã được tính toán và các điểm ảnh có giá trị lớn nhất được xem là nằm trên xương của đối tượng.

Chương 2: KỸ THUẬT PHÁT HIỆN XƯƠNG NHỜ PHÉP TOÁN HÌNH THÁI

2.1. Các phép toán hình thái và các tính chất

Hiểu một cách đầy đủ thì “Morphology” là hình thái học và cấu trúc của đối tượng, hay nó mô tả những phạm vi và các mối quan hệ giữa các thành phần trong một đối tượng. Hình thái học quá quen thuộc trong các lĩnh vực ngôn ngữ học và sinh học. Trong ngôn ngữ học, hình thái học là sự nghiên cứu về cấu trúc của từ, tập hợp từ, câu... và đó cũng là một lĩnh vực nghiên cứu nhiều năm nay. Còn trong sinh học, hình thái học lại chú trọng tới hình dạng của một cá thể hơn, chẳng hạn có thể phân tích hình dạng của một chiếc lá để từ đó có thể nhận dạng được đó là loại cây gì; nghiên cứu hình dạng của một nhóm vi khuẩn, dựa trên các đặc điểm nhận dạng để phân biệt chúng thuộc nhóm vi khuẩn nào... Tùy theo trường hợp cụ thể mà có cách phân lớp phù hợp với nó: Có thể phân lớp dựa trên những hình dạng của mặt cắt như (Elip, tròn...), kiểu và mức độ của những hình dạng bất quy tắc (lồi, lõm, ...), những cấu trúc trong (lỗ, đường thẳng, đường cong, ...) mà đã được tích lũy qua nhiều năm quan sát.

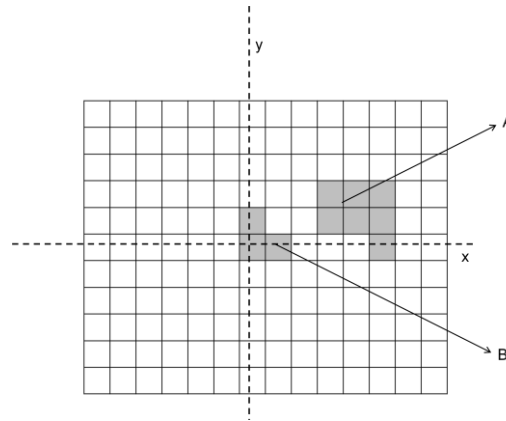
Tính khoa học của Hình thái học số mới chỉ thực sự phát huy khả năng của nó kể từ khi máy tính điện tử số ra đời và làm cho hình thái học trở lên thông dụng, có nhiều tính năng mới. Những đối tượng trong Hình thái học ta có thể coi như là tập hợp của các điểm ảnh, nhóm lại theo cấu trúc ma trận hai chiều. Những thao tác toán học rời rạc trên tập hợp điểm đó được sử dụng để làm rõ những nét đặc trưng riêng của hình dạng đối tượng, do vậy có thể tính toán được hay nhận biết được chúng một cách dễ dàng. Phần lớn các phép toán hình thái học được định nghĩa từ hai phép toán cơ bản là phép toán co nhị phân (Erosion) và phép toán giãn nhị phân (Dilation) .

Phép toán cơ sở được kết hợp với một đối tượng là tiêu chuẩn của các phép toán tập hợp như phép hợp (Union), Phép giao (InterSection), và phép bù (Complement) cộng với phép tịnh tiến nào đó. Vì vậy trong phần tiếp theo sẽ trình bày các khái niệm về tập hợp thường được sử dụng trong phép toán hình thái.

2.1.1. Khái niệm cơ bản

Ta đã định nghĩa ảnh nhị phân là tập hợp các điểm ảnh có tọa độ $a(x,y)$. Chúng ta còn có định nghĩa khác về ảnh, theo quan sát thì có thể xem ảnh như tập hợp các tọa độ rời rạc hoặc liên tục.

Theo một định nghĩa nào đó thì, tập hợp này tương ứng với các điểm ảnh thuộc về các đối tượng hiện hữu trong ảnh.



Hình 2.1: Ảnh và đối tượng trong ảnh.

Hình vẽ cho thấy hai đối tượng, hay hai tập hợp A và B trong ảnh. Ở đây ta cần phải xác định hệ trục tọa độ như trong hình, quan tâm đến giá trị các điểm ảnh cấu thành lên đối tượng trong ảnh và được giới hạn trên không gian rời rạc \mathbb{Z}^2 .

Đầu tiên, ta có A là một tập hợp trong không gian rời rạc \mathbb{Z}^2 . Nếu $a = (a_1, a_2)$ là một phần tử của tập hợp A thì ta có thể viết như sau:

$$a \in A \quad (0.1)$$

Như thế, nếu a không phải là một phần tử của tập hợp A thì ta viết:

$$a \notin A \quad (0.2)$$

Tập hợp mà không có phần tử gọi là tập hợp rỗng hoặc tập hợp trống và được biểu thị bằng ký tự \emptyset .

Tập hợp A được xác định bởi nội dung nằm trong 2 dấu ngoặc nhọn $\{ \}$.

Các phần tử của tập hợp có liên quan đến ở trong chương này là những tọa độ của các pixel đại diện cho đối tượng hoặc các đặc trưng khác được quan tâm trong một hình ảnh.

Thí dụ, khi ta viết một công thức:

$$C = \{ w \mid w = -d, d \in D \} \quad (0.3)$$

Tập hợp C là tập hợp gồm các phần tử w , như vậy w được hình thành khi nhân từng tọa độ của các phần tử trong tập hợp D với -1 .

Nếu mỗi phần tử của tập hợp A cũng là một phần tử trong tập hợp B thì ta có thể nói A là tập con của B , và được biểu thị như sau:

$$A \subseteq B, \quad (0.4)$$

Hợp (Union) của hai tập hợp A và B là tập hợp của các phần tử thuộc A , B hoặc cả A và B , được biểu thị như sau:

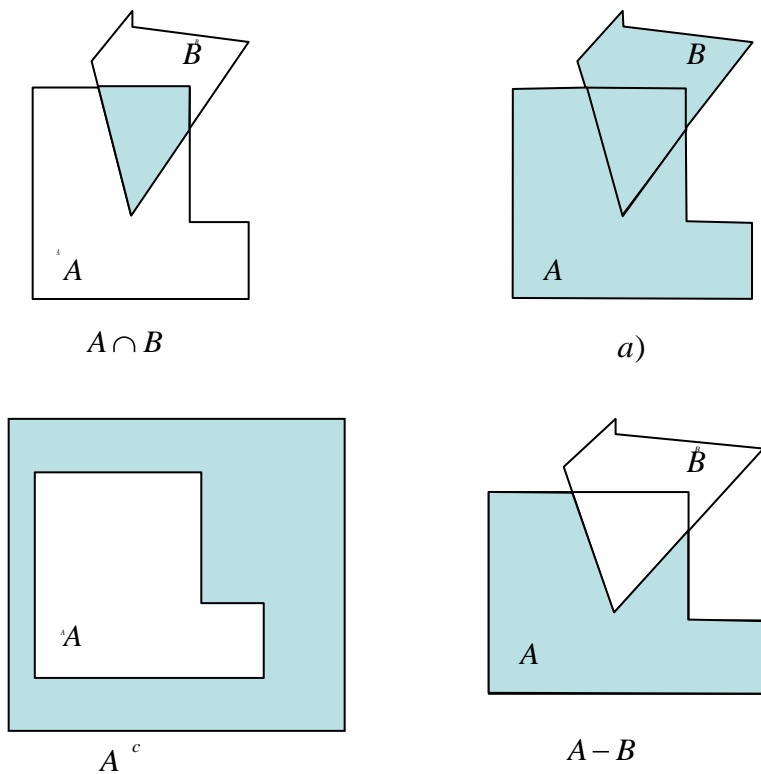
$$C = A \cup B, \quad (0.5)$$

Như vậy giao (Intersection) của hai tập hợp A và B là tập hợp các phần tử thuộc cả A và B , được biểu thị như sau:

$$D = A \cap B, \quad (0.6)$$

Hai tập hợp A và B trở thành phân chia (Disjoint) hoặc loại trừ lẫn nhau (Mutually exclusive) nếu chúng có chung các phần tử, như trong trường hợp sau:

$$A \cap B = \emptyset, \quad (0.7)$$



Hình 2.2: Quan hệ giữa hai tập hợp

a) Giao giữa tập hợp A và tập hợp B ; b) Hợp giữa tập hợp A và tập hợp B ; c) Phần bù tuyệt đối của A ; d) Phần bù tương đối của tập hợp A trong tập hợp B (Phép trừ hai tập hợp A và B).

Phần bù tuyệt đối (complement) của một tập hợp A là tập hợp các phần tử không bao gồm các phần tử trong A :

$$A^c = \{w | w \notin A\}, \quad (0.8)$$

Phần bù tương đối (difference) của tập hợp A trong tập hợp B được biểu thị là $A - B$ như trong biểu thức sau:

$$A - B = \{w \mid w \in A, w \notin B\} = A \setminus B, \quad (0.9)$$

Với w là tập hợp các phần tử thuộc tập hợp A nhưng không thuộc tập hợp B , khái niệm này được minh họa trên Hình 2.2(d). Phần có màu là kết quả của phép toán giữa hai tập hợp.

Sau đây là hai định nghĩa cần bổ sung vì nó được sử dụng rộng rãi trong phép toán hình thái, nhưng nói chung là vẫn chưa tìm được cái căn bản trên thuyết tập hợp.

Phản xạ (reflection) của tập hợp B được biểu thị là \hat{B} , và được định nghĩa bằng biểu thức sau:

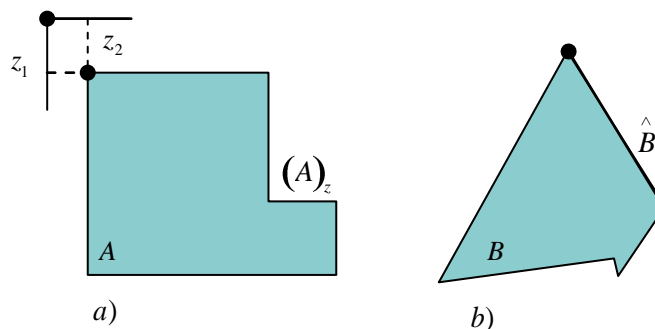
$$\hat{B} = \{w \mid w = -b, b \in B\}, \quad (0.10)$$

Nghĩa là phần tử w được hình thành khi nhân từng tọa độ của tập hợp B với -1 khi đó vị trí tương đối giữa các điểm trong tập hợp B có xu hướng đối ngược lại so với ban đầu, gọi là tính phản xạ của tập hợp. Hay nói cách khác tập hợp B sẽ quay quanh gốc tọa độ một góc 180° .

Một điểm $z(z_1, z_2)$ thuộc tập hợp A , khi z tịnh tiến thì phép tịnh tiến của tập hợp A được biểu thị là A_z , và được định nghĩa bằng biểu thức sau:

$$A_z = \{c \mid c = a + z, a \in A\}, \quad (0.11)$$

Khi một điểm z có tọa độ $z(z_1, z_2)$ hình thành trong tập hợp A tịnh tiến một lượng là z thì tập hợp A cũng tịnh tiến theo hướng của điểm z với một lượng tương đương. Hình 2.3 cho thấy hai định nghĩa trên và sử dụng hai tập hợp A và B ở Hình 2.3, chấm màu đen trên hình vẽ là để biểu thị phản xạ của tập hợp B và sự tịnh tiến của 1 điểm từ tập hợp A . (Some basic concept from set theory/ Digital image processing Second Edition, tr 520~521)



Hình 2.3: Phép biến đổi của tập hợp

a) Tịnh tiến của tập hợp A từ một điểm z ; b) Phản xạ của tập hợp B .

Lưu ý: Những phần tử riêng lẻ hợp thành B không chỉ có các điểm ảnh mà còn có các vector khi chúng có vị trí tọa độ xác định với điểm gốc $[0,0]$.

Bảng 2.1: Ba phép toán logic cơ bản:

p	q	$p \wedge q$	$p \vee q$	$\neg p$
0	0	0	0	1
0	1	0	1	1
1	0	0	1	0
1	1	1	1	0

2.1.1.1. Phép giãn nhị phân (Dilation)

Tập hợp B thường thì được coi như là một phần tử cấu trúc(structuring element) trong giãn nhị phân, cũng như trong các phép toán hình thái khác, tập hợp A là tập hợp các phần tử của hình ảnh gốc.

Với A và B là các tập hợp trong \mathbb{Z}^2 , thì phép giãn nhị phân của A theo B ($A \oplus B$) được định nghĩa qua công thức sau:

$$A \oplus B = \left\{ z \mid \left(\hat{B} \right)_z \cap A \neq \emptyset \right\}, \quad (0.12)$$

Như vậy phép giãn nhị phân của tập hợp A bởi phần tử cấu trúc B là tập hợp của tất cả các điểm z (z là tâm điểm của phần tử cấu trúc B trên tập hợp A) sao cho phần xạ của B_z giao với tập A tại ít nhất một điểm. Hay nói cách khác, phép giãn nhị phân là sự chồng chéo từ ít nhất một phần tử từ phần xạ của phần tử cấu trúc B với tập hợp A . Đồng thời các phần tử này phải là tập con của tập hợp A .

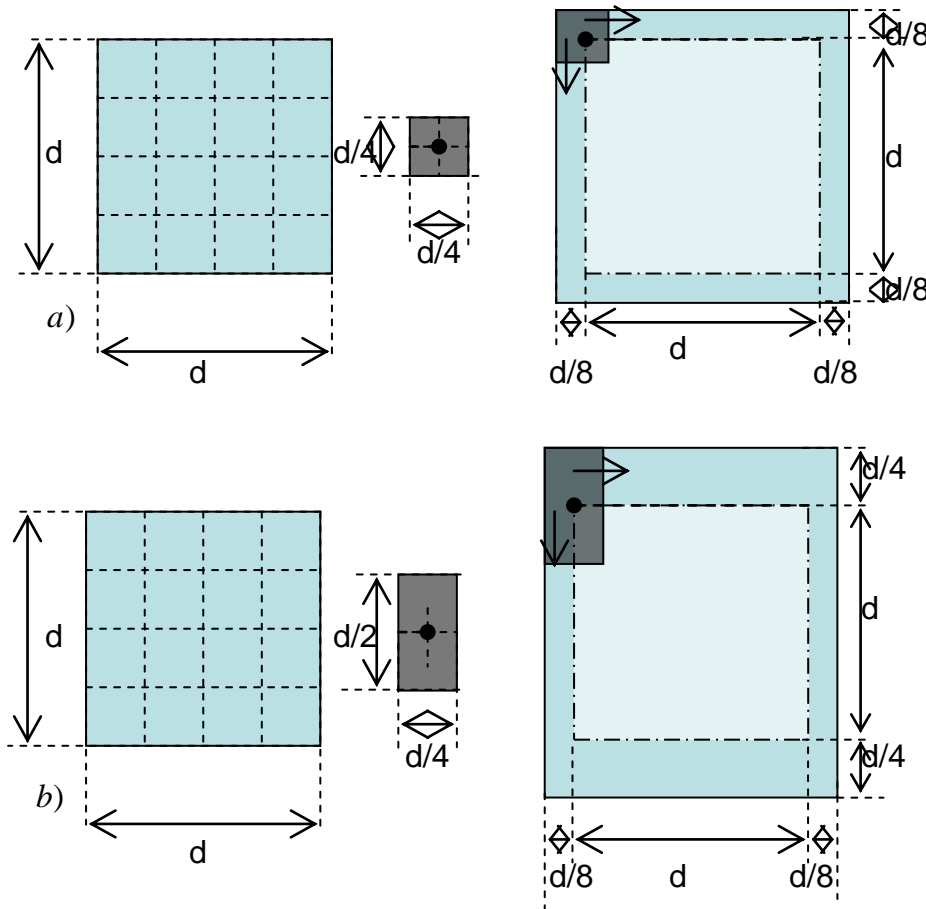
Công thức (1.12) có thể được viết lại :

$$A \oplus B = \left\{ z \mid \left[\left(\hat{B} \right)_z \cap A \right] \subseteq A \right\}, \quad (0.13)$$

Phép giãn nhị phân của tập hợp A bởi tập hợp B là tồn tại các điểm w thuộc \mathbb{Z}^2 sao cho w là tổng của hai điểm tương ứng bất kỳ thuộc tập hợp A và tập hợp B , định nghĩa này được mô tả qua công thức :

$$A \oplus B = \{w \in Z^2 \mid w = a + b, a \in A, b \in B\}, \quad (0.14)$$

Tổng quát hơn, nếu A là một hình ảnh và B là phần tử cấu trúc có tâm điểm nằm trên hình ảnh A , khi đó phép giãn của hình ảnh A bởi phần tử cấu trúc B có thể được hiểu như quỹ tích của các điểm được phủ bởi phần tử cấu trúc B khi tâm điểm của B di chuyển trên cạnh của hình ảnh A .



Hình 2.1: Phép giãn nhị phân.

Hình 2.6a gồm:

- + Tập hợp A có hai cạnh bên kích thước là d .
- + Phần tử cấu trúc vuông B kích thước $d/4$, trường hợp này thì phần tử cấu trúc B và tương phản của nó bằng nhau vì B có tập hợp các phần tử đối xứng nhau qua tâm điểm (đấu chấm đen ở giữa).
- + Cuối cùng là kết quả của phép giãn nhị phân giữa tập hợp A và phần tử cấu trúc B .

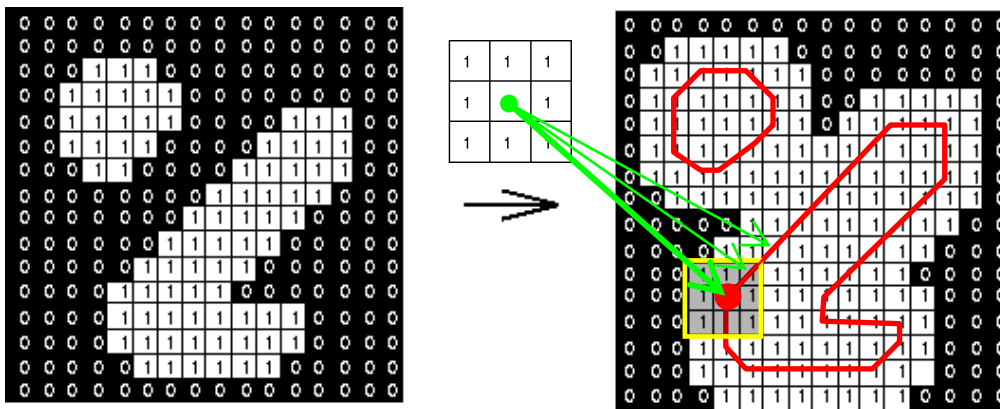
Hình 2.6b cũng gồm những thành phần tương tự nhưng với phần tử cấu trúc B là hình chữ nhật, nhưng cho ta một kết khác. Như vậy, mỗi kiểu phần tử cấu trúc khác nhau sẽ cho ta một kết quả khác nhau, sau khi thuật toán được thi hành.

Từ đó ta có công thức:

$$A \oplus B = \bigcup_{b \in B} A_b, \quad (0.15)$$

Để thực tế hóa những lý thuyết đã nêu trên, ta coi những phần tử cấu trúc như một mẫu sẵn và dịch chuyển tịnh tiến trên bề mặt hình ảnh. Khi gốc của phần tử cấu trúc (chấm đen ở tâm điểm) khớp với điểm ảnh tại cạnh của hình ảnh thì các điểm ảnh tương ứng với với gốc này sẽ được đánh dấu và thay thế. Sau khi toàn bộ điểm ảnh đã được quét qua bởi phần tử cấu trúc thì, thao tác giãn hình ảnh được hoàn tất.

Một ví dụ về phép giãn trên một hình ảnh nhị phân sử dụng phần tử cấu trúc dạng ma trận vuông 3×3 như sau:



Hình 2.2: Quá trình quét của phần tử cấu trúc trên hình ảnh nhị phân.

Ở ví dụ trên ta các điểm ảnh màu trắng mang giá trị là 1 là các điểm thuộc đối tượng đang cần quan tâm trên ảnh, và phần màu đen mang giá trị 0 là phần nằm ngoài đối tượng. Khi thuật toán được thi hành thì phần tử cấu trúc sẽ lần lượt quét qua các điểm ảnh ngoài cùng (Đi theo đường kẻ màu đỏ trên hình vẽ) của đối tượng sau đó thay thế các điểm ảnh trên đối tượng này theo mẫu phần tử cấu trúc. Từ đó ta ứng dụng để nối các nét bị đứt gãy của đoạn văn do quá trình xuống cấp, với khoảng cách lớn nhất của các nét bị đứt gãy tầm hai điểm ảnh.

2.1.1.2. Phép co nhị phân (Erosion)

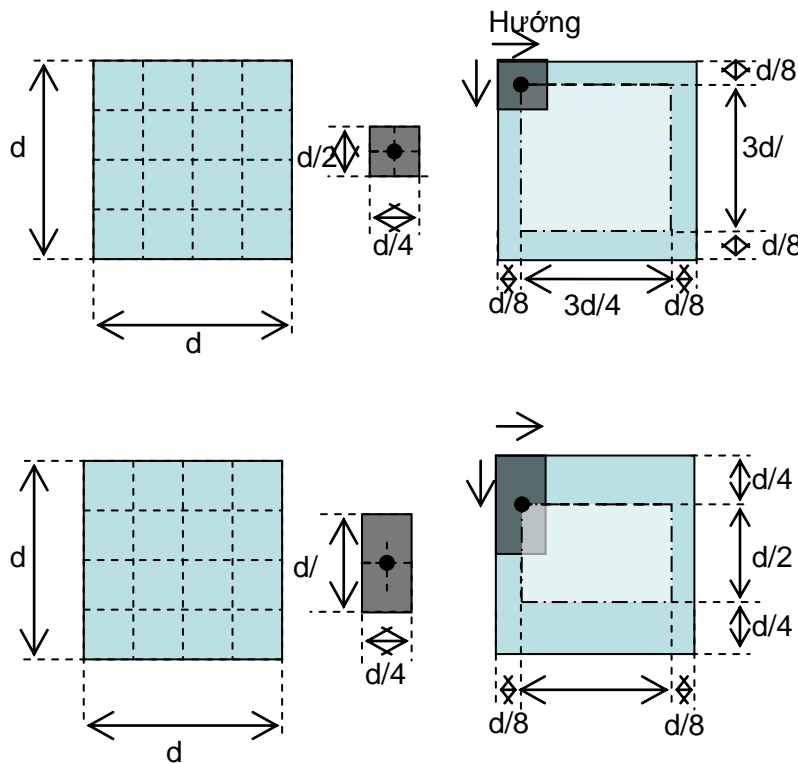
Ta cũng xét tập hợp A và tập hợp B (Phần tử cấu trúc) trong \mathbb{Z}^2 , thì phép co nhị phân của tập hợp A bởi phần tử cấu trúc B được kí hiệu $A \ominus B$ và viết dưới dạng công thức như sau:

$$A \ominus B = \{z \mid B_z \subseteq A\}, \quad (0.16)$$

$$\text{Với } B_z = b + z, b \in B, \quad (0.17)$$

Phép co nhị phân của tập hợp A bởi phần tử cấu trúc B là tập hợp các điểm z (z nằm ở tâm điểm của phần tử cấu trúc B) sao cho B_z là tập con của A .

Xét hình vẽ sau:



Hình 2.3: Phép co nhị phân trên hai đối tượng.

Hình 2.8a gồm:

- + Tập hợp A có hai cạnh bên kích thước là d .
- + Phần tử cấu trúc vuông B kích thước $d/4$ (Dấu chấm đen ở giữa là tâm điểm).
- + Cuối cùng là kết quả của phép co nhị phân giữa tập hợp A và phần tử cấu trúc B .

Phần có màu nhạt hơn là kết quả sau khi thực hiện co hình ảnh bởi phần tử cấu trúc B . Hình 2.8b gồm những thành phần tương tự nhưng với phần tử cấu trúc B là hình chữ nhật, và cho ta một kết quả khác.

Vậy phép co nhị phân của ảnh A với phần tử cấu trúc B là quỹ tích các điểm được tạo ra bởi tâm điểm của phần tử cấu trúc B khi tịnh tiến trên hình ảnh A .

Từ đó ta có công thức:

$$A \ominus B = \bigcap_{b \in B} A_{-b}, \quad (0.18)$$

Phép co nhị phân và giãn nhị phân có thể được với nhau qua phép bù và phép phản xạ của tập hợp, luận lý này sẽ được minh họa qua công thức sau:

$$(A \ominus B)^c = A^c \oplus \hat{B}, \quad (0.19)$$

Ta chứng minh công thức trên là đúng:

Từ công thức co nhị phân ta có:

$$(A \ominus B)^c = \{z \mid B_z \subseteq A^c\}, \quad (0.20)$$

Nếu tập hợp B_z là tập con tập hợp A thì ta có, $B_z \cap A^c = \emptyset$, cho nên, trong trường hợp này ta sẽ có:

$$(A \ominus B)^c = \{z \mid B_z \cap A^c = \emptyset\}, \quad (0.21)$$

Vì phần bù của phép giãn nhị phân giữa tập hợp A và tập hợp B luôn thỏa mãn:

$$(B_z \cap A^c = \emptyset)^c = B_z \cap A^c \neq \emptyset$$

cho nên:

$$(A \ominus B)^c = \{z \mid B_z \cap A^c \neq \emptyset\}$$

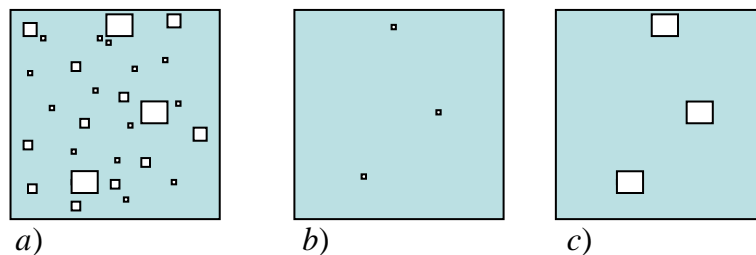
Mặt khác theo công thức 1.12 ta có:

$$A \oplus B = \left\{ z \mid \left(\hat{B} \right)_z \cap A \neq \emptyset \right\}$$

Suy ra: $(A \ominus B)^c = A^c \oplus \hat{B}$, \square .

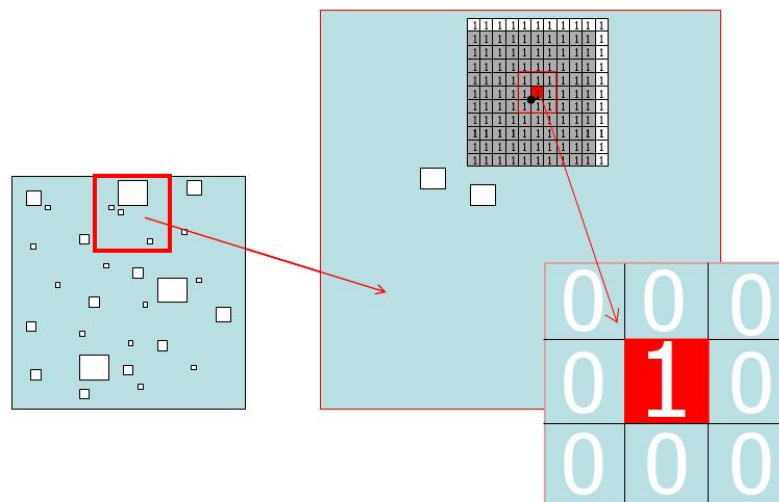
Như vậy, phần bù của phép co nhị phân giữa tập hợp A và Phần tử cấu trúc B là phép giãn nhị phân giữa phần bù của tập hợp A với phần xạ của phần tử cấu trúc B .

Một ứng dụng quan trọng của phép co nhị phân là dùng để loại trừ các chi tiết không cần thiết trên hình ảnh. Thí dụ, Trên một hình ảnh, ta có các đối tượng có cỡ tương ứng 1,4,6 và 11 điểm ảnh, Bây giờ nếu muốn loại trừ các đối tượng nhỏ không cần thiết trên ảnh, chỉ để lại các đối tượng có kích thước lớn, như trong hình vẽ đối tượng ta cần giữ lại là những đối tượng có kích thước 11 điểm ảnh. Ta sẽ sử dụng phần tử cấu trúc có kích thước 10x10 điểm ảnh để thực hiện phép co nhị phân (Erosion). Kết quả sẽ chỉ còn lại 3 đối tượng có kích thước 1 điểm ảnh(Hình 2.9b). Sau đó để các đối tượng trở lại kích thước ban đầu ta sử dụng phép giãn nhị phân(Dilation) với phần tử cấu trúc có kích cỡ tương ứng (Hình 2.9c).



Hình 2.4: Quá trình lọc đối tượng sử dụng phép co nhị phân và phép giãn nhị phân.

Quá trình thực hiện có thể được minh họa rõ ràng qua hình vẽ sau:



Hình 2.5: Ứng dụng của phép co ảnh dưới dạng số nhị phân.

a) Hình ảnh ban đầu; b) Hình ảnh quá trình co nhị phân trên đối tượng với phần tử cấu trúc 10x10, phần tử được tô đậm màu sẽ có giá trị 1 sau quá trình co nhị phân; c) Phóng to đối tượng và giá trị của đối tượng sau quá trình co nhị phân với phần tử cấu trúc 10x10.

2.1.2. Một số tính chất của phép toán hình thái

Để thuận tiện trong việc cài đặt ta có thể phân tích và đơn giản hóa các mẫu phức tạp, dưới đây là một số tính chất cơ sở của phép toán hình thái.

2.1.2.1. Các mệnh đề

-Mệnh đề 1: Tính chất gia tăng

$$\begin{aligned} \text{a)} \quad X \subseteq X' &\Rightarrow X \ominus B \subseteq X' \ominus B, & \forall B \\ &X \oplus B \subseteq X' \oplus B, & \forall B \end{aligned}$$

$$\begin{aligned} \text{b)} \quad B \subseteq B' &\Rightarrow X \ominus B \supseteq X' \ominus B, & \forall X \\ &X \oplus B \subseteq X \oplus B', & \forall X \end{aligned}$$

-Mệnh đề 2: Tính chất phân phối với phép \cup

$$\begin{aligned} \text{a)} \quad X \oplus (B \cup B') &= (X \oplus B) \cup (X \oplus B'), \\ \text{b)} \quad X \ominus (B \cup B') &= (X \ominus B) \cap (X \ominus B'), \end{aligned}$$

- Mệnh đề 3: Tính chất phân phối với phép giao

$$(X \cap Y) \ominus B = (X \ominus B) \cap (Y \ominus B),$$

- Mệnh đề 4: Tính chất kết hợp

$$\begin{aligned} \text{a)} \quad (X \oplus B) \oplus B' &= X \oplus (B \oplus B'), \\ \text{b)} \quad (X \ominus B) \ominus B' &= X \ominus (B \oplus B'), \end{aligned}$$

2.1.2.2. Định lý

Giả sử, X là một đối tượng trong ảnh, B là phần tử cấu trúc, khi đó X sẽ bị chặn trên và chặn dưới bởi tập close của X theo B.

$$\text{Ta có, } (X \oplus B) \ominus B \supseteq X \supseteq (X \ominus B) \oplus B,$$

Chứng minh,

Ta có: $\forall x \in X \Rightarrow B_x \subseteq X \oplus B,$ (Vì $X \oplus B = \bigcup_{x \in X} B_x$)

$\Rightarrow x \in (X \oplus B) \ominus B$ (Theo định nghĩa của phép co),

$\Rightarrow (X \oplus B) \ominus B \supseteq X,$ (0.22)

Mặt khác, $\forall y \in (X \ominus B) \oplus B,$

Suy ra: $\exists x \in X \ominus B$ sao cho, $y \in B_x$ (Vì $(X \ominus B) \oplus B = \bigcup_{x \in X \ominus B} B_x$)

$\Rightarrow B_x \in X \Rightarrow y \in X,$

Suy ra: $X \supseteq (X \ominus B) \oplus B,$ (0.23)

Từ (1.22) và (1.23) ta có:

$(X \oplus B) \ominus B \supseteq X \supseteq (X \ominus B) \oplus B, \square$

2.1.2.3. Hệ quả

Tính Bất biến:

a) $((X \oplus B) \ominus B) \oplus B = (X \oplus B),$

b) $((X \ominus B) \oplus B) \ominus B = (X \ominus B),$

Thật vậy, từ định lý trên ta có: $X \subseteq (X \oplus B) \ominus B,$

Do đó, theo tính chất gia tăng ta có:

$\Rightarrow X \oplus B \subseteq ((X \oplus B) \ominus B) \oplus B, \quad (0.24)$

Mặt khác cũng theo định lý trên ta có: $(X \ominus B) \oplus B \subseteq X \quad \forall X$

$\Rightarrow ((X \ominus B) \oplus B) \ominus B \subseteq X \ominus B, \quad (0.25)$

Từ (1.24) và (1.25) ta có:

$\Rightarrow ((X \oplus B) \ominus B) \oplus B = X \oplus B. \square$

Và $((X \ominus B) \oplus B) \ominus B = X \ominus B. \square$

(Phát triển kỹ thuật dò biên, phát hiện biên và ứng dụng/Tr. 15)

2.2. Kỹ thuật tìm xương nhờ phép toán hình thái

2.2.1. Trích biên (Boundary Extraction)

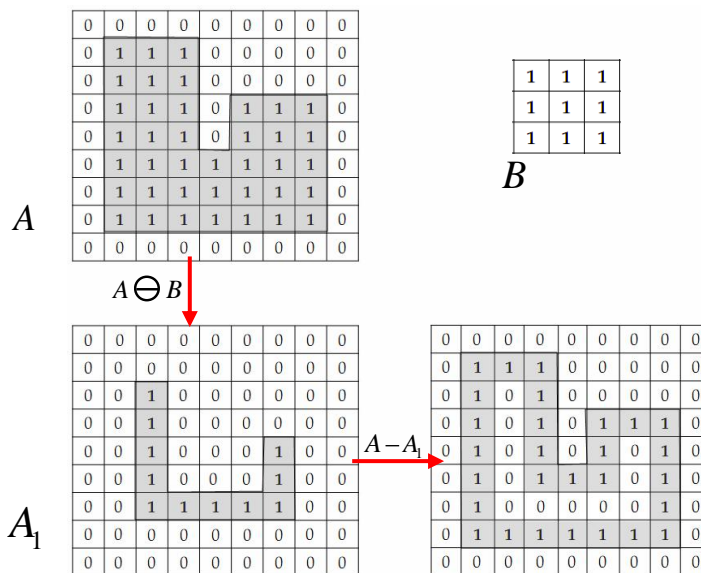
Với ảnh đầu vào là ảnh xám, thì ta có thể xử lý phân đoạn ảnh bằng các ngưỡng của ảnh. Trong hình ảnh mỗi điểm ảnh có giá trị xám riêng, giá trị này được giới hạn trong khoảng 0 tới 255. Vì vậy ta có thể thông qua lược đồ màu để lựa chọn các mức xám thích hợp. Khi chuyển sang ảnh nhị phân các điểm ảnh có giá trị lớn hơn ngưỡng này đều được gán là 255 hoặc nếu không thì sẽ được gán là 0. ngược lại những điểm ảnh nào có giá trị nằm trong ngưỡng đều được đặt là 1, những điểm ảnh này là điểm ảnh cấu thành đối tượng trong ảnh nhị phân.

Trong ảnh nhị phân, đối tượng được cấu thành bởi các điểm ảnh liên thông có giá trị là 1, Xét một thí dụ ở hình 2.16 có 9 điểm ảnh dọc, 9 điểm ảnh ngang. Những điểm ảnh tạo lên biên là những điểm ảnh thuộc đối tượng đó nhưng trong 8 điểm lân cận phải có ít nhất một điểm ảnh có giá trị là 0.

Biên của tập hợp A phụ thuộc vào kích thước của phần tử cấu trúc. Độ dày của đường viền bao quanh đối tượng phụ thuộc vào kích thước của phần tử cấu trúc. Cho 1 thí dụ, nếu một phần tử cấu trúc có kích thước 3x3 sinh ra độ dày đường viền là 1 thì với phần tử cấu trúc có kích thước 5x5 sẽ sinh ra đường viền của đối tượng có độ dày là 3.

Khi đó, biên của tập hợp A, ký hiệu là ∂A , được xác định bởi công thức:

$$\partial A = A - (A \ominus B), \quad (1.33)$$



Hình 2.6: Quá trình tìm biên của đối tượng trên ảnh nhị phân.

2.2.2. Làm đầy (Region Filling)

Làm đầy vùng của một đối tượng từ biên là bổ xung giá trị 1 vào toàn bộ vùng ở phía bên trong biên của đối tượng. Cho một ảnh nhị phân A , các điểm ảnh ở biên của đối tượng trong hình ảnh A có giá trị là 1 và các điểm ảnh khác có giá trị là 0. Ta bắt đầu quá trình làm đầy bằng cách gán giá trị 1 vào điểm ảnh p ở bên trong biên của đối tượng, sau đó lặp đi lặp lại phép giãn nhị phân giữa điểm ảnh với phần tử cấu trúc B dưới giới hạn đặt ra bởi phép bù của tập hợp A (A^c). Bởi vì nếu hạn chế không được đặt thì chương trình sẽ lặp đi lặp lại phép gán này, dẫn tới sảy ra hiện tượng tràn qua các cùng khác trên ảnh, thậm chí toàn bộ ảnh.

Với $X_0=p$ và B là phần tử cấu trúc, phép làm đầy vùng trong ảnh sẽ được xác định bởi công thức:

$$X_k=(X_{k-1} \oplus B) \cap A^c, \text{ Với } k=1,2,3,\dots,k-1. \quad (1.34)$$

Thuật toán sẽ dừng khi $X_k=X_{k-1}$.

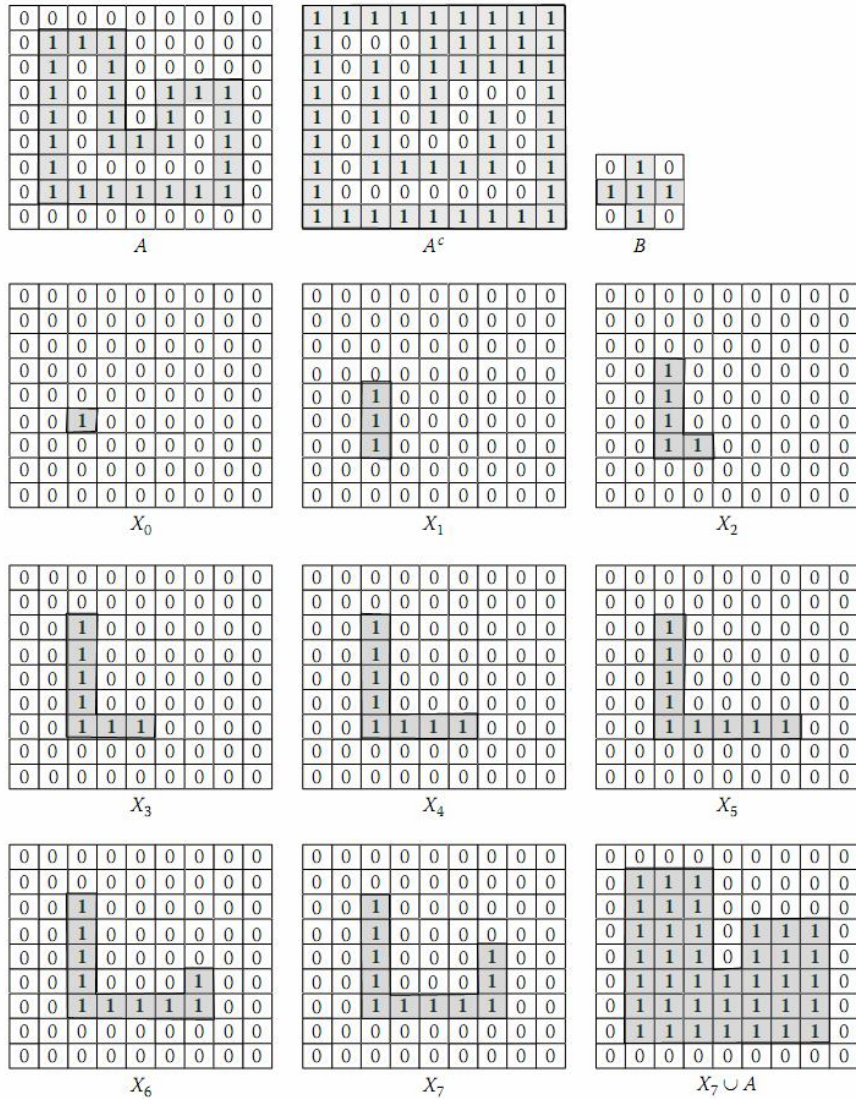
Thí dụ, ở hình 2.17, A là ảnh ban đầu chỉ chứa biên của đối tượng (giá trị là 1), B là phần tử cấu trúc có kích thước 3×3 và hình dạng chữ thập.

Khi đó thuật toán sẽ thực hiện lần lượt theo trình tự như sau:

Bước 1: Thuật toán sẽ gán giá trị 1 cho một điểm ảnh p bên trong vùng biên.

Bước 2: Áp dụng phép giãn nhị phân từ điểm ảnh p với phần tử cấu trúc B , đồng thời lấy giao của kết quả phép giãn với phần bù của tập hợp A . Như vậy, sau khi lấy giao của hai tập hợp thì phần nào có giá trị là 1 tương xứng với giá trị 1 của phần bù tập hợp A thì được giữ nguyên, phần nào có giá trị 1 tương xứng với giá trị 0 của phần bù tập hợp A thì sẽ được gán là 0. Quá trình này sẽ được lặp đi lặp lại cho đến khi tất cả các điểm ảnh tương ứng bên trong biên của đối tượng được gán là 1.

Bước 3: Cuối cùng lấy hợp giữa X_7 và biên A ta sẽ được đối tượng đã được làm đầy từ biên vùng trong ảnh (Hình 2.17).



Hình 2.7: Quá trình làm đầy đối tượng trong ảnh.

2.2.3. Tách các thành phần liên thông (Extraction of Connected Components)

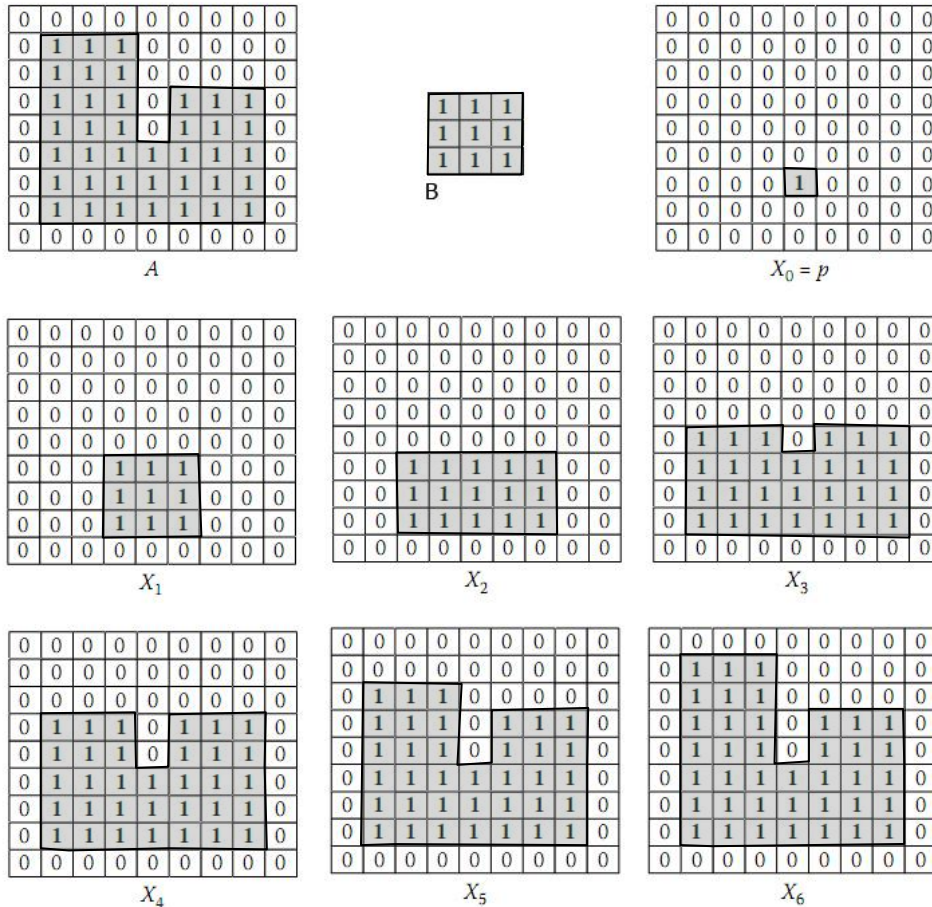
Trong thực tế, bài toán tách thành phần liên thông trong ảnh nhị phân đóng vai trò quan trọng trong các ứng dụng phân tích ảnh tự động. Tương tự như phép làm đầy vùng. Quá trình thực hiện thuật toán cũng được lặp đi lặp lại với phép tách các thành phần liên thông.

Cho một ảnh nhị phân trong đó đối tượng là tập hợp A bao gồm một số thành phần liên thông. Những điểm ảnh nào thuộc đối tượng thì được đánh dấu là 1 và điểm ảnh nào nằm ngoài đối tượng thì được đánh dấu là 0. Ta có thể tách các thành phần liên thông

bằng cách gán một điểm ảnh ở trong đối tượng rồi từ điểm ảnh đó, lặp đi lặp lại phép giãn nhị phân với phần tử cấu trúc dưới hạn chế đặt ra bởi tập hợp A.

Với $X_0=p$ và B là phần tử cấu trúc. Khi đó phép tách các thành phần liên thông được xác định bởi công thức:

$$X_k = (X_{k-1} \oplus B) \cap A, \quad k=1,2,3\dots \quad (1.35)$$



Hình 2.8: Quá trình tách các thành phần liên thông trong ảnh bằng phép toán hình thái.

2.2.4. Bao Lồi (Convex Hull)

Ta có thể hình dung bao lồi của một đối tượng giống như một dải băng cao su bao quanh đối tượng. Vì thế dải băng cao su này được căng qua các phần lồi theo các viền lồi của đối tượng. Kết quả là tạo ra một hình dạng đối tượng không có đoạn nào bị lõm. Bao lồi có thể được biến đổi bởi phép toán Hit-or-Miss với 4 phần tử cấu trúc luân phiên

quay 90^0 . Cho A là một ảnh nhị phân và phần tử cấu trúc B^i với $i=1,2,3,4$, khi đó ta có 4 phần tử cấu trúc tương ứng như sau:

$$B^1 = \begin{bmatrix} 1 & \times & \times \\ 1 & 0 & \times \\ 1 & \times & \times \end{bmatrix}, \quad B^2 = \begin{bmatrix} 1 & 1 & 1 \\ \times & 0 & \times \\ \times & \times & \times \end{bmatrix}, \quad B^3 = \begin{bmatrix} \times & \times & 1 \\ \times & 0 & 1 \\ \times & \times & 1 \end{bmatrix}, \quad B^4 = \begin{bmatrix} \times & \times & \times \\ \times & 0 & \times \\ 1 & 1 & 1 \end{bmatrix}, \quad (1.36)$$

Trong các phần tử cấu trúc trên những phần tử được đánh dấu \times là những phần tử không cần quan tâm. Đầu tiên cho $X_0^i = A$, sau đó lặp đi lặp lại thuật toán theo công thức sau:

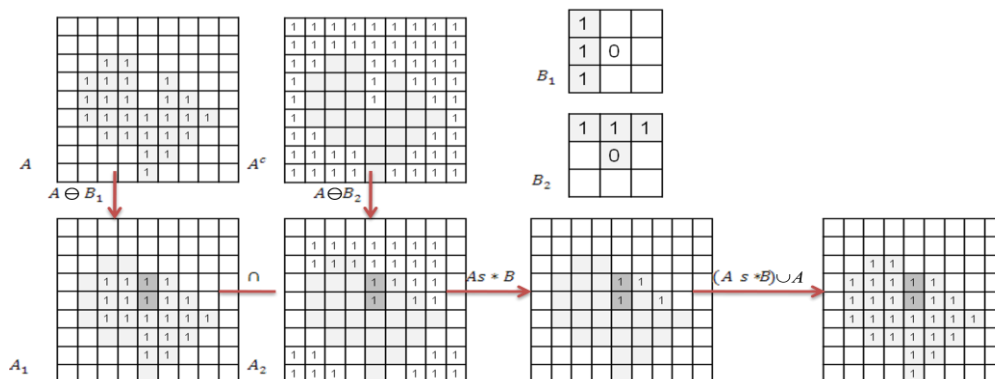
$$X_k^i = (X_{k-1}^i \circ B^i) \cup A, \quad i=1,2,3,4 \text{ và } k=1,2,3 \dots k-1 \quad (1.37)$$

Khi $X_k^i = X_{k-1}^i$ tức là chúng đồng quy (converge) thì quá trình xử lý dừng lại và đặt $D^i = X_{conv}^i$, Khi đó bao lồi của A sẽ là:

$$C(A) = \bigcup_{i=1}^4 D^i, \quad (1.38)$$

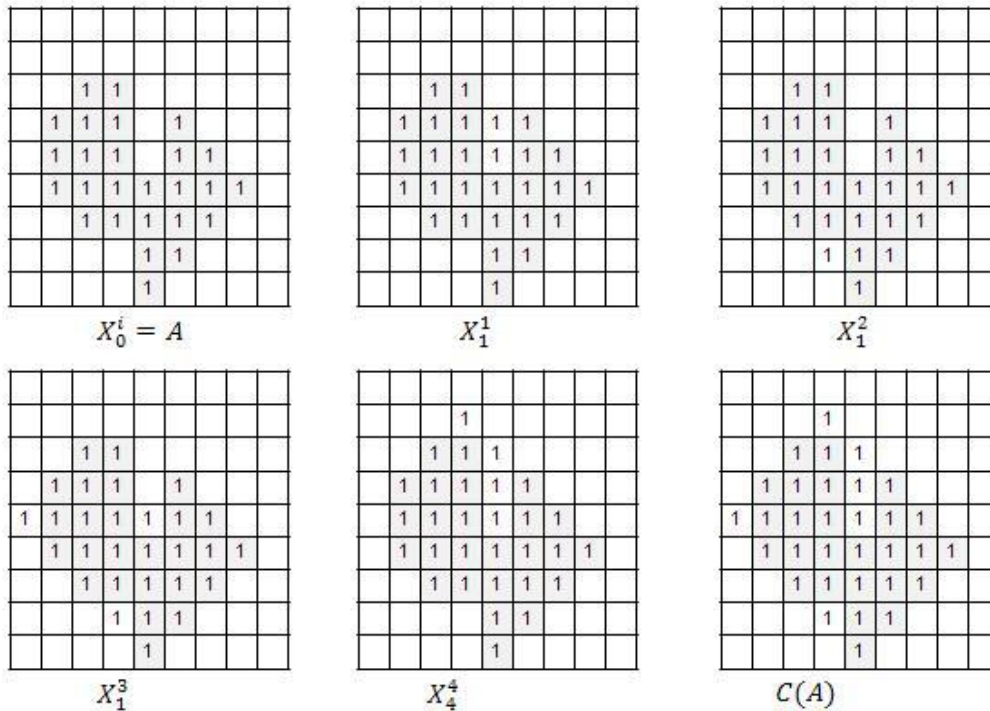
Nói cách khác, thuật toán sẽ thực hiện phép biến đổi Hit-or-Miss giữa tập hợp A với phần tử cấu trúc B^i trên toàn bộ đối tượng cho đến khi đối tượng không có gì thay đổi nữa thì dừng lại và đặt $D^i = X_{conv}^i$, sau đó hợp tập hợp A với kết quả D^i , quá trình này sẽ được thực hiện lần lượt với B^2, B^3, B^4 . Cuối cùng hợp của 4 tập D^i cho ta bao lồi của A .

Xét thí dụ trong một chu trình xử lý ảnh sử dụng phép bao lồi (Hình 2.19). Với tập hợp A là tập hợp các phần tử tạo lên một đối tượng trong ảnh, A^c là phần bù của tập hợp A . B là phần tử cấu trúc gồm có B_1 , và B_2 .



Hình 2.9: Một quá trình trong phép bao lồi.

Thí dụ trên cho thấy một chu trình trong quá trình thực hiện phép bao lồi đối với đối tượng trong ảnh. Quá trình này sẽ được lặp đi lặp lại lần lượt với các phần tử cấu trúc khác nhau (1.37). Khi thuật toán kết thúc, tất cả các chỗ bị lõm hay thiếu hụt chi tiết sẽ được làm đầy. Xét một thí dụ tổng quát từ thí dụ trong hình 2.19, với $D^1 = X_1^1, D^2 = X_1^2, D^3 = X_1^3$, và $D^4 = X_1^4$:



Hình 2.10: Phép bao lồi đối với đối tượng trong ảnh.

2.2.5. Làm mảnh(Thinning)

Thuật toán làm mảnh thường bao gồm nhiều lần lặp, mỗi lần lặp tất cả các điểm ảnh của đối tượng sẽ được kiểm tra, Các phần tử cấu trúc được thiết kế để tìm các điểm biên mà khi loại bỏ điểm ảnh trên đối tượng thì sẽ không làm ảnh hưởng tới các liên thông. Nếu điểm ảnh nào mà thỏa mãn điều kiện của phần tử cấu trúc thì sẽ bị loại bỏ. Quá trình này cứ lặp đi lặp lại cho tới khi không còn điểm biên nào được xóa nữa.

Làm mảnh tập hợp A bởi phần tử cấu trúc B , ký hiệu $A \otimes B$, xác định bởi công thức sau:

$$\begin{aligned} A \otimes B &= A - (A s * B) \\ &= A \cap (A s * B)^c, \quad (1.39) \end{aligned}$$

Để có thể làm mảnh một đối tượng A một cách hiệu quả hơn, ta dựa vào dãy các phần tử cấu trúc.

$$\{B\} = \{B^1, B^2, \dots, B^n\}, \quad (1.40)$$

Trong đó B^i Nhận được từ B^{i-1} , qua phép quay quanh tâm điểm của nó. Với khái niệm này ta định nghĩa phép làm mảnh bởi một dãy các phần tử cấu trúc theo công thức.

$$A \otimes \{B\} = (((A \otimes B^1) \otimes B^2) \dots) \otimes B^n, \quad (1.41)$$

Nói cách khác, quá trình làm mảnh A bởi B^1 , kết quả lại tiếp tục được làm mảnh bởi B^2 cho đến B^n . Lúc này đối tượng trong ảnh hầu như không thay đổi nữa.

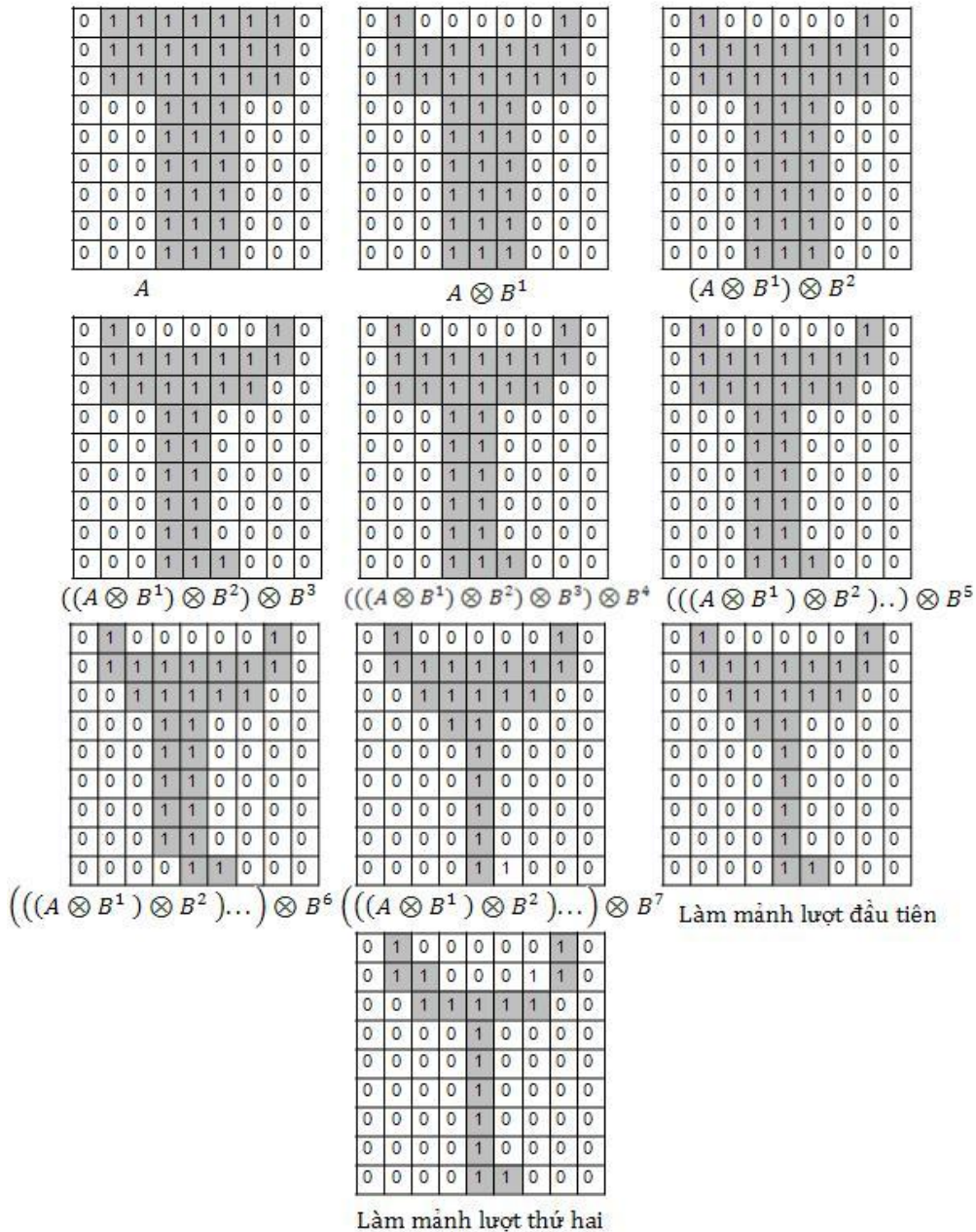
Xét thí dụ quá trình làm mảnh (Hình 2.21). Quá trình làm mảnh đối tượng sẽ được lặp đi lặp lại cho đến khi đối tượng không có sự thay đổi nào thì dừng lại .

$$\begin{aligned} B^1 &= \begin{bmatrix} 0 & 0 & 0 \\ \times & 1 & \times \\ 1 & 1 & 1 \end{bmatrix}, & B^2 &= \begin{bmatrix} \times & 0 & 0 \\ 1 & 1 & 0 \\ 1 & 1 & \times \end{bmatrix}, & B^3 &= \begin{bmatrix} 1 & \times & 0 \\ 1 & 1 & 0 \\ 1 & \times & 0 \end{bmatrix}, & B^4 &= \begin{bmatrix} 1 & 1 & \times \\ 1 & 1 & 0 \\ \times & 0 & 0 \end{bmatrix}, \\ B^5 &= \begin{bmatrix} 1 & 1 & 1 \\ \times & 1 & \times \\ 0 & 0 & 0 \end{bmatrix}, & B^6 &= \begin{bmatrix} \times & 1 & 1 \\ 0 & 1 & 1 \\ 0 & 0 & \times \end{bmatrix}, & B^7 &= \begin{bmatrix} 0 & \times & 1 \\ 0 & 1 & 1 \\ 0 & \times & 1 \end{bmatrix}, & B^8 &= \begin{bmatrix} 0 & 0 & \times \\ 0 & 1 & 1 \\ \times & 1 & 1 \end{bmatrix}. \end{aligned}$$

Theo công thức tổng quát ta có:

$$A \otimes \{B\} = (((A \otimes B^1) \otimes B^2) \dots) \otimes B^8,$$

Như vậy, Trong thí dụ này quá trình làm mảnh được lặp đi lặp lại 2 lượt. Trong mỗi lần lặp đối tượng phải trải qua quá trình làm mảnh với 8 phần tử cấu trúc: $B^1, B^2, B^3, \dots, B^8$.



Hình 2.11: Quá trình làm mảnh đối tượng trong hình ảnh.

2.2.6. Tìm khung xương (Skeletonization)

Tìm khung xương cũng tương tự như làm mảnh, nhưng nó được dùng để tìm ra các chi tiết cấu trúc của đối tượng. Khung xương của đối tượng là tập hợp các điểm ảnh cách đều biên của đối tượng. Và có thể biểu diễn bằng phép co nhị phân và phép mở ảnh.

Xét A là một ảnh nhị phân bao gồm các điểm ảnh thuộc đối tượng, được đặt nhãn là các số 1. Các điểm ảnh không thuộc đối tượng được đặt nhãn là các số 0. B là phần tử cấu trúc 3×3 . Khi đó, nếu ký hiệu $S(A)$ là khung xương của tập hợp A thì thuật toán tìm khung xương được xác định qua công thức:

$$S(A) = \bigcup_{k=0}^K S_k(A), \quad (1.42)$$

Trong đó: $S_k(A) = \bigcup_{k=0}^k \{(A \ominus kB) - [(A \ominus kB) \circ B]\}, \quad (1.43)$

$(A \ominus kB)$ là phép co k lần liên tiếp trên A . Tức là:

$$(A \ominus kB) = ((\dots (A \ominus B) \dots) \ominus B \dots) \ominus B$$

Và K là Bước lặp cuối cùng trước khi A bị ăn mòn thành tập rỗng. Nói cách khác:

$$K = \max\{k \mid (A \ominus kB) \neq \emptyset\}$$

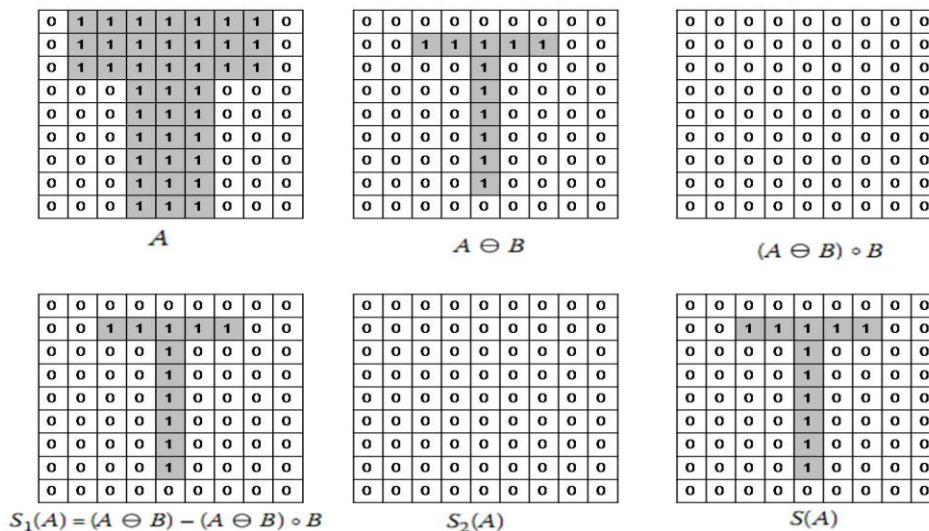
Công thức (1.42) và (1.43) khẳng định rằng khung xương $S(A)$ có thể nhận được từ hợp các bộ khung xương con $S_k(A)$. Có thể chứng minh rằng A có thể xây dựng lại từ các tập con này bằng cách sử dụng công thức:

$$A = \bigcup_{k=0}^K (S_k A \oplus kB), \quad (1.44)$$

Trong đó $(S_k A \oplus kB)$ ký hiệu k lần giãn nhị phân $S_k(A)$, tức là:

$$(S_k A \oplus kB) = ((\dots (S_k(A) \oplus B) \dots) \oplus B \dots) \oplus B.$$

Xét thí dụ:



Hình 2.12: Quá trình thực hiện thuật toán tìm xương.

Chương 3: CHƯƠNG TRÌNH THỬ NGHIỆM

3.1. Bài toán

Do tính chất công việc cũng như thẩm mỹ của con người luôn luôn thay đổi để phù hợp với thực tiễn mà những đòi hỏi, yêu cầu đặt ra cho xử lý ảnh ngày càng cao, đa dạng. Theo xu hướng đó, xử lý ảnh phát triển không ngừng hướng tới quy trình xử lý ảnh hoàn thiện.

Với bài toán đặt ra, em cài đặt chương trình thử nghiệm

- Đầu vào : Một ảnh bất kỳ
- Đầu ra : Ảnh đã được xử lý

3.2. Phân tích và giới thiệu

3.2.1. Phân tích

Hoạt động của chương trình :

Bước 1: Đưa một ảnh có định dạng JPG (hoặc PNG, BMP, GIF, JPEG).

Bước 2: Chương trình quét và xử lý.

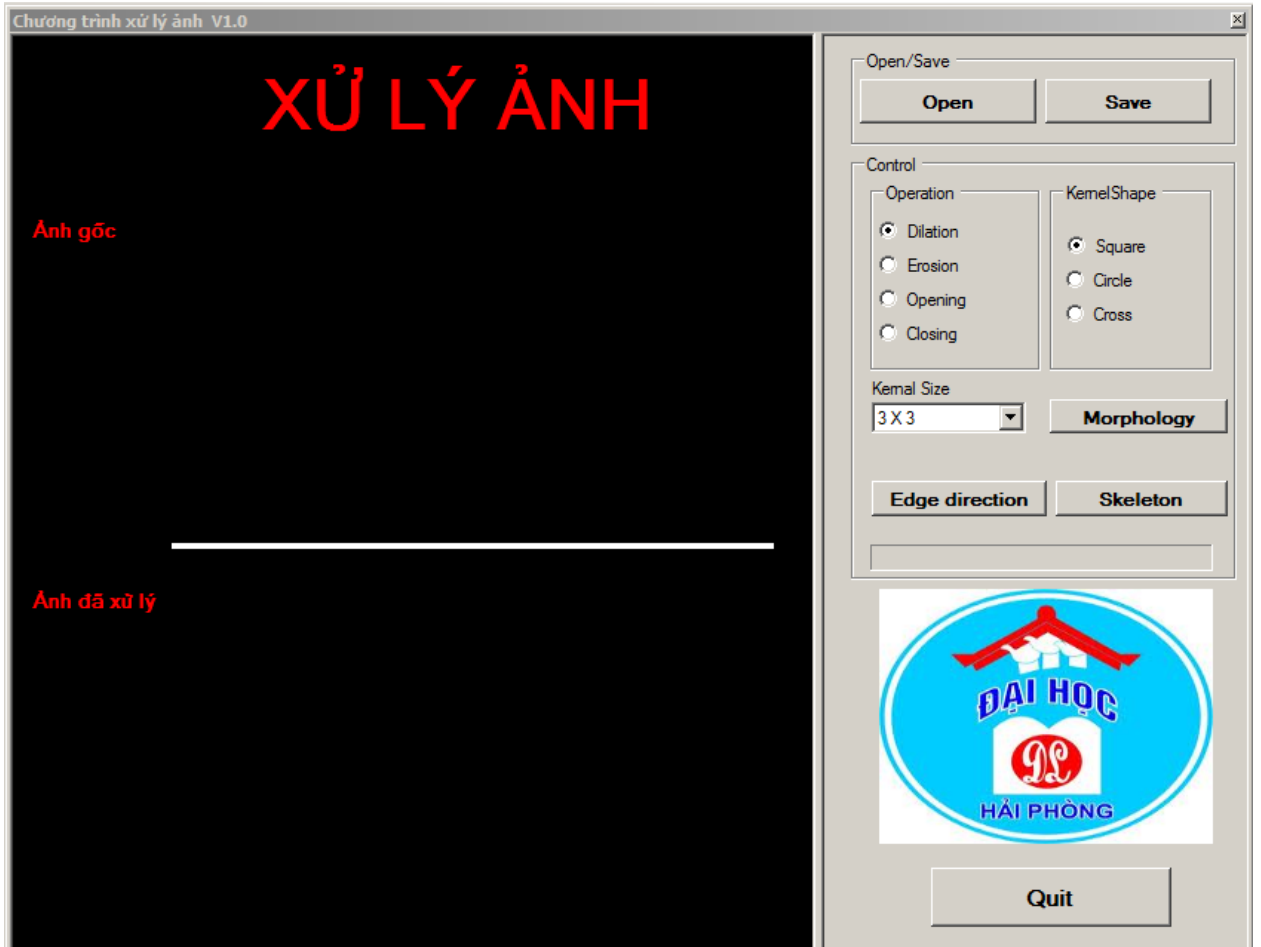
Bước 3: Xử lý dựa vào các phép toán hình thái.

Bước 4: Hiện thị ảnh kết quả.

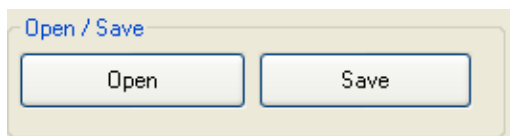
3.2.2. Giới thiệu và hướng dẫn sử dụng

Chương trình được xây dựng trên nền C#, để minh họa các phép toán hình thái gồm có, phép co nhị phân (Erosion), phép giãn nhị phân (Dilation), phép đóng ảnh (Closing), phép mở ảnh (Opening), tìm biên (Edge direction), và tìm xương (Skeleton).

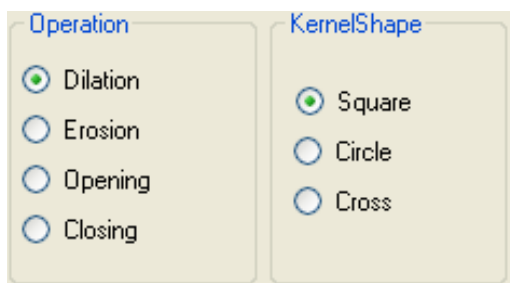
Giao diện chương trình như sau:



Hướng dẫn sử dụng

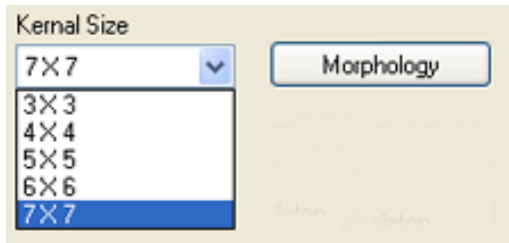


Open: Nạp ảnh cần xử lý vào picture box
Save: Lưu ảnh đã xử lý ra định dạng Image file.



Ở nhóm Operation: Lựa chọn các phép toán để tác động lên ảnh gốc. Phép giãn nhị phân (Dilation), phép co nhị phân (Erosion), Phép mở ảnh (Opening), Phép đóng ảnh (Closing).

Nhóm “Kernel Shape”: Lựa chọn các dạng của phần tử cấu trúc, tác động lên đối tượng trong ảnh. Dạng hình vuông (Square), dạng hình tròn (Circle), và dạng hình dấu cộng (Cross).



“Kernel Size”: Dùng để lựa chọn kích thước của phần tử cấu trúc.

Morphology: Thực hiện phép toán hình thái tác động lên ảnh khi đã chọn phép toán và dạng cỡ phần tử cấu trúc tương xứng.



Edge direction: Phát hiện biên của đối tượng trong ảnh.

Skeleton: Bắt đầu quá trình làm mảnh đối

tượng trong ảnh.



Status bar: Thanh trạng thái biểu thị tiến độ xử lý ảnh của chương trình. Cho một thí dụ, nếu sử dụng phép co nhị phân hay giãn nhị phân, thì

tiến độ này sẽ chạy một lần, bởi vì để thực hiện phép co nhị phân và phép giãn thì ta chỉ thực hiện thuật toán một lần. Nếu sử dụng với phép đóng ảnh và phép mở ảnh thì sẽ chạy hai lần, vì để thực hiện một trong hai phép toán này ta phải thực hiện hai lần thuật toán co và giãn nhị phân. Đối với các phép toán khác thì chức năng biểu thị trạng thái của nó cũng tương tự.

3.3. Một số kết quả của chương trình

Chương trình được cài đặt thuật toán để xử lý ảnh nhị phân bằng các phép toán phép giãn nhị phân, phép co nhị phân, phép đóng ảnh, và phép mở ảnh với các phần tử cấu trúc dạng hình vuông, hình tròn, hình chữ thập, có kích thước khác nhau (3x3, 4x4, 5x5, 6x6, 7x7).

Quan sát hình 4.1 ta thấy: Từ một ảnh nhị phân đã xuống cấp, sau khi sử dụng phép giãn nhị phân với phần tử cấu trúc dạng tròn, kích thước 4x4, ta được ảnh mới,



Hình 3.1: Thực nghiệm giãn nhị phân hình ảnh với cấu trúc 4x4.

Thí dụ như vẫn trường hợp trên, ta xét thuật toán giãn nhị phân trên ảnh với phần tử cấu trúc 7x7 ta sẽ có ảnh:



Hình 3.2: Thực hiện giãn nhị phân hình ảnh với phần tử cấu trúc 7x7

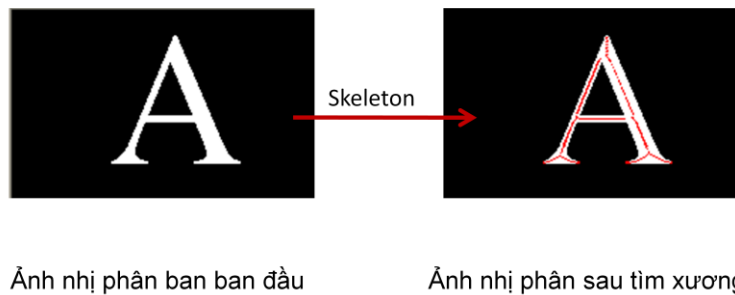
Cũng hình ảnh nhị phân như trên ta sử dụng phép co nhị phân hình ảnh với phần tử cấu trúc 4x4. Bởi vì với phép toán này đối tượng có xu hướng co vào nên ta có thể sử dụng để lọc các điểm ảnh nhỏ thừa trong ảnh nhị phân. Vì vậy tùy theo yêu cầu mà ta tùy chọn kích thước phần tử cấu trúc tương ứng để lọc các chi tiết quan trọng trên ảnh.

Quan sát ở hình 4.3 ta có thể thấy, với phần tử cấu trúc 4x4, khi ta cài đặt thuật toán và tác động lên ảnh thì tập hợp những điểm ảnh có kích thước lớn sẽ không bị loại bỏ bởi phần tử cấu trúc, trong khi các tập hợp điểm ảnh nhỏ hoặc điểm ảnh riêng rẽ sẽ bị loại bỏ. Như trên phần lý thuyết ở chương 2 đã giới thiệu, trong một số trường hợp ta có thể kết hợp phép co nhị phân và phép giãn nhị phân để lọc các điểm ảnh không dư thừa trong ảnh. Tất nhiên chỉ đạt yêu cầu đối với ảnh nhị phân.



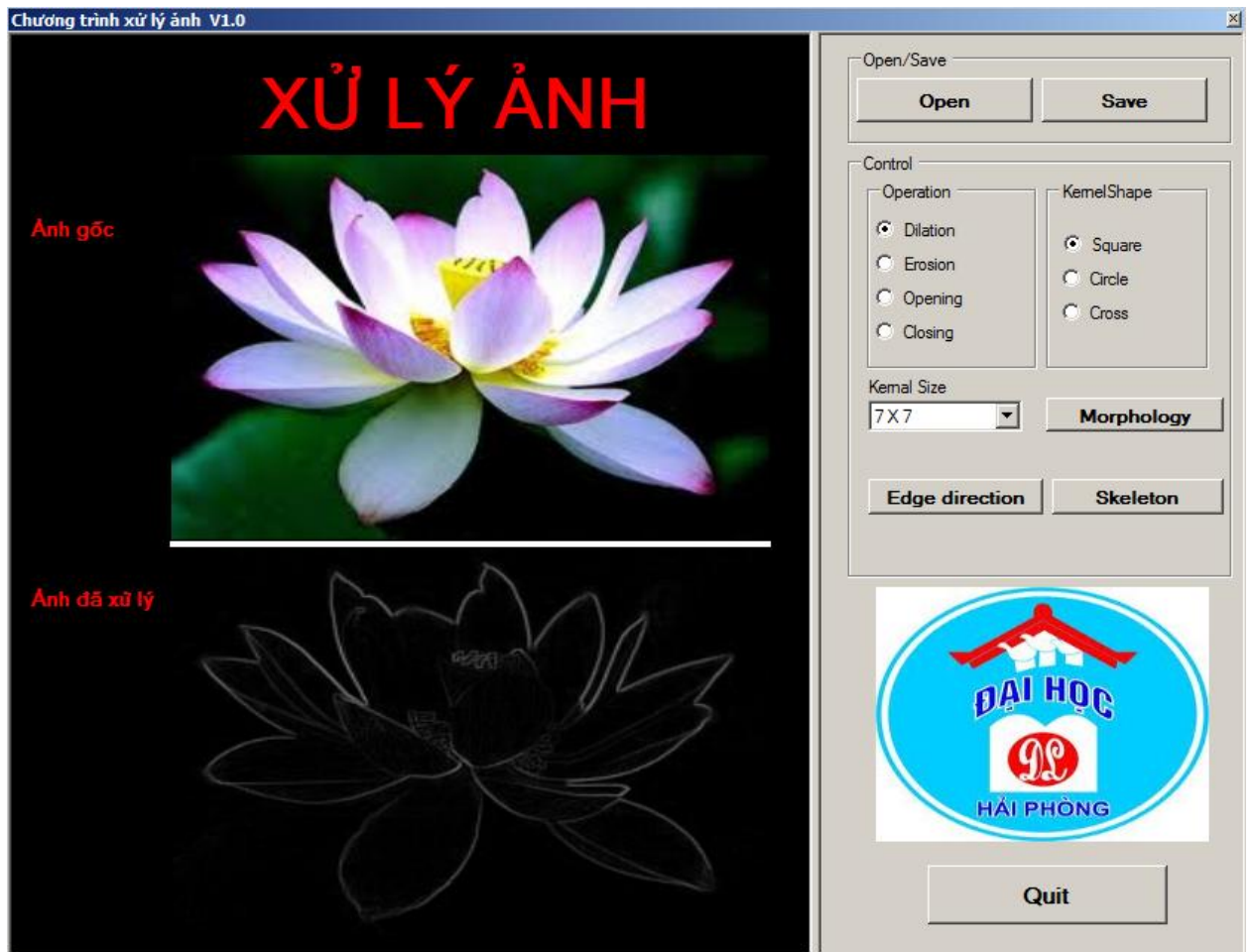
Hình 3.3: Thực nghiệm phép co nhị phân trên hình ảnh với phần tử cấu trúc 4x4.

Tiếp theo, ta xét một hình ảnh nhị phân đối tượng là chữ A màu trắng (Hình 4.9), ta áp dụng thuật toán tìm xương trên hình ảnh này. Kết quả thu được là tập hợp các điểm màu đỏ phía bên trong A.



Hình 4.9: Thực nghiệm tìm xương đối tượng trên ảnh nhị phân.

Cuối cùng là thuật toán phát hiện biên đối tượng. Đầu tiên thuật toán sẽ thực hiện công việc phân giữa đối tượng và phần tử cấu trúc, sau đó lấy đối tượng ban đầu trừ đi kết quả công việc phân, ta sẽ được biên của đối tượng, độ dày của biên tùy thuộc vào phần tử cấu trúc.



Hình 4.5: Thực nghiệm tìm biên của đối tượng.

PHẦN KẾT LUẬN

Phép toán hình thái là một mảng vô cùng quan trọng trong xử lý ảnh số, các đề tài về phép biến đổi này cũng đang được nhiều người quan tâm, bởi vì qua các thuật toán và phép toán cơ sở đó ta có thể mở rộng và phát triển những bài toán khác có liên quan đến lĩnh vực quan sát thăm dò, trí tuệ nhân tạo..., chẳng hạn như ứng dụng với camera theo dõi qua các thuật toán nhận biết đối tượng chuyển động, hoặc ghép đối tượng trong hai hình ảnh thông qua các đặc điểm hoặc điểm tương đồng...

Các kết quả đã đạt được:

- Trình bày tổng quan về xử lý ảnh, lịch sử phát triển và phương thức hoạt động.
- Trình bày hệ thống các khái niệm cơ bản của phép toán hình thái dưới góc độ xử lý ảnh; các khái niệm, các tính chất và ứng dụng cơ bản trong xử lý và biểu diễn hình dạng đối tượng. Đồng thời qua đó giải thích và minh họa cụ thể và đưa ra mối liên hệ giữa các phép toán đó với nhau.
- Thực nghiệm các phép toán, thuật toán qua chương trình xử lý.

Qua phân tích trên ta có thể rút ra kết luận:

Các phép toán giãn nhị phân (Dilation), co nhị phân (Erosion), Phép mở ảnh (Opening), phép đóng ảnh (Closing), là các phép toán cơ sở trong phép toán hình thái, các thuật toán trích biên (Boundary), làm đầy (Region filling), tách thành phần liên thông (Extraction of connected components), bao lồi (Convex Hull), làm mảnh (Thinning), tìm xương (Skeleton), đều được xây dựng từ các phép toán hình thái cơ bản. Các phép toán đó là điều kiện, cũng như tiền đề để tìm hiểu nghiên cứu các thuật toán khác có tính ứng dụng cao trong xử lý ảnh nói riêng và trong thực tế đời sống nói chung.

Phép toán hình thái là một đề tài rất rộng, cần rất nhiều thời gian mới có thể tìm hiểu và nghiên cứu về nó. Trong thời gian tới em sẽ tiếp tục tìm hiểu các thuật toán lọc nhiễu, tăng cường ảnh, và phân đoạn ảnh... thông qua các phép toán hình thái.

Vì vậy em rất mong nhận được sự đóng góp ý kiến quý báu của các thầy cô giáo cũng như bạn bè để báo của em được hoàn thiện hơn.

Em xin chân thành cảm ơn!

TÀI LIỆU THAM KHẢO

Tiếng Việt

- [1]. TS. Đỗ Năng Toàn, TS. Phạm Việt Bình (2007) - Giáo Trình Môn Học Xử Lý Ảnh, trường ĐH thái nguyên, khoa CNTT, tr. 26 ~ 27, tr. 44 ~ 49.
- [2]. Lương Mạnh Bá, Nguyễn Thanh Thủy (2003)- Nhập Môn Xử Lý Ảnh Số, Nhà xuất bản Khoa học và Kỹ thuật, tr. 7 ~ 10.
- [3]. Phạm Việt Bình (2006) – Phát triển kỹ thuật dò biên, phát hiện biên và ứng dụng, luận án tiến sĩ khoa học, tr. 5, tr. 13~17
- [4]Đào Bảo Linh – C5Tin – ĐH Điện Lực HN “Tìm hiểu các phép toán hình thái và ứng dụng” đồ án tốt nghiệp ngành C5Tin
- [5] Phan Hữu Mạnh- CT701-CNTT- “Tìm hiểu các phép toán hình thái và ứng dụng” đồ án tốt nghiệp ngành CNTT
-

Tiếng Anh

- [1]. Frank Y. Shih(2009) -Image Processing and Mathematical Morphology Fundamentals and Applications, tr. 11~ 24,tr 37~52.
- [2]. J.B.T.M Roerdink- Computer vision And Mathematical Morphology.