



ĐẠI HỌC QUỐC GIA TP. HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN

GIÁO TRÌNH

XỬ LÝ ẢNH

Biên soạn: ThS. Võ Đức Khánh
Hiệu đính: GS.TSKH. Hoàng Kiếm



NHÀ XUẤT BẢN
ĐẠI HỌC QUỐC GIA TP HỒ CHÍ MINH

**ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN**

Giáo trình

XỬ LÝ ẢNH

Tác giả: ThS Võ Đức Khánh

Hiệu đính: GS-TSKH Hoàng Văn Kiếm

**NHÀ XUẤT BẢN ĐẠI HỌC QUỐC GIA
TP HỒ CHÍ MINH - 2008**

LỜI NÓI ĐẦU

Giáo trình này là một trong các giáo trình chính yếu của chuyên ngành Công nghệ thông tin. Giáo trình được xây dựng theo phương châm vừa đáp ứng yêu cầu chuẩn mực của sách giáo khoa, vừa có giá trị thực tiễn, đồng thời tăng cường khả năng tự học, tự nghiên cứu của sinh viên. Trên cơ sở đó, chúng tôi đã tham khảo nhiều tài liệu có giá trị của các tác giả trong và ngoài nước và đã sử dụng nhiều ví dụ lấy từ các ứng dụng thực tiễn.

Giáo trình này được dùng kèm giáo trình điện tử trên đĩa CD trong đó có thêm phần trình bày của giảng viên, các bài tập và phần đọc thêm nhằm đáp ứng tốt nhất cho việc tự học của sinh viên.

Chúng tôi rất mong nhận được các ý kiến đóng góp để giáo trình ngày càng hoàn thiện.

Nhóm biên soạn

MỤC LỤC

CHƯƠNG 1: KHÁI QUÁT VỀ XỬ LÝ ẢNH	11
1.1. Giới thiệu chung	11
1.2. Các bước cơ bản trong xử lý ảnh	12
1.3. Tài nguyên học tập	18
CHƯƠNG 2: ẢNH SỐ HÓA & SỰ CẢM NHẬN THỊ GIÁC ...	21
2.1. Các giá trị thông dụng	22
2.2. Những đặc trưng của các phép toán trên ảnh	23
2.2.1. Kiểu phép toán trên ảnh	23
2.2.2. Kiểu lân cận	24
2.3. Các tham số video	26
2.4. Sự cảm nhận tri giác	27
2.5. Độ nhạy sáng	28
2.5.1. Độ nhạy với bước sóng	28
2.5.2. Độ nhạy với tác nhân kích thích	28
2.6. Độ nhạy với tần số không gian	30
2.7. Độ nhạy về màu sắc	31
2.7.1. Bộ quan sát chuẩn	31
2.7.2. Các tọa độ màu CIE	32
2.8. Các áo giác quang học	35
CHƯƠNG 3: LẤY MẪU ẢNH	37
3.1. Độ trừ mật lấy mẫu trong xử lý ảnh	38
3.1.1. Cửa chụp lấy mẫu	38
3.2. Độ trừ mật lấy mẫu trong phân tích ảnh	40
3.2.1. Lấy mẫu để đo diện tích	40
3.2.2. Lấy mẫu để đo độ dài	41
3.2.3. Kết luận về lấy mẫu	42

CHƯƠNG 4: CÁC LOẠI NHIỄU	44
4.1. Nhiễu photon	44
4.2. Nhiễu nhiệt	46
4.3. Nhiễu điện từ trên chip	47
4.4. Nhiễu KTC	47
4.5. Nhiễu khuếch đại	48
4.6. Nhiễu lượng hóa	18
CHƯƠNG 5: CÁC THIẾT BỊ THU ẢNH VÀ HIỂN THỊ ẢNH ...	50
5.1. Tính tuyến tính	50
5.2. Độ nhạy	51
5.2.1. Độ nhạy tuyệt đối	51
5.2.2. Độ nhạy tương đối	52
5.3. SNR	54
5.3.1. Nhiễu nhiệt (dòng tối)	54
5.3.2. Nhiễu photon	55
5.4. Tạo bóng	56
5.5. Dạng pixel	56
5.5.1. Các pixel vuông	57
5.5.2. Hệ số làm đầy	58
5.6. Độ nhạy phổ	59
5.7. Tốc độ chớp (thời gian tích hợp)	60
5.7.1. Tốc độ chớp trên các video camera	60
5.7.2. Tốc độ chớp trên các camera khoa học	60
5.8. Tốc độ đọc ra	61
5.9. Các thiết bị hiển thị ảnh	62
5.9.1. Tốc độ làm tươi	62
5.9.2. Hệ số đan xen	62
5.9.3. Độ phân giải	63

CHƯƠNG 6: CÁC PHÉP TOÁN XỬ LÝ ẢNH	64
6.1. Phép toán chập	64
6.2. Các tính chất của phép toán chập	64
6.3. Các biến đổi Fourier	65
6.4. Các tính chất của biến đổi Fourier	66
6.4.1. Tầm quan trọng của độ lớn và pha	69
6.4.2. Các tín hiệu đối xứng vòng	71
6.4.3. Ví dụ về các tính hiệu 2 – chiều phổ biến cùng với các biến đổi tương ứng của chúng	72
6.5. Các mô tả thống kê	75
6.5.1. Hàm phân phối xác suất về độ sáng	75
6.5.2. Hàm trừ mật xác suất về độ sáng	76
6.5.3. Đại lượng trung bình	78
6.5.4. Độ lệch chuẩn	78
6.5.5. Hệ số biến đổi	79
6.5.6. Đại lượng phần trăm	79
6.5.7. Tỷ số tín hiệu – nhiễu	80
6.6. Biểu diễn biên của đối tượng	82
6.6.1. Mã xích	82
6.6.2. Các tính chất của mã xích	82
6.6.3. Mã crack	84
6.6.4. Mã chạy	85
6.7. Các phép toán trên lược đồ	85
6.7.1. Phép co giãn độ tương phản	85
6.7.2. Cân bằng lược đồ	86
6.7.3 Các phép toán dựa trên lược đồ khác	88
6.8. Các phép toán toán học	88
6.8.1. Các thao tác nhị phân	88
6.8.2. Các thao tác số học	91

6.9. Các phép toán dựa trên phép chập	91
6.9.1. Kiến thức nền	91
6.9.2. Phép chập trong miền tọa độ	94
6.9.3. Phép chập trong miền tần số	98
6.10. Các phép toán làm trơn ảnh	99
6.10.1. Các bộ lọc tuyến tính	99
6.10.2. Các bộ lọc phi tuyến	106
6.10.3. Tóm tắt các thuật toán làm trơn	108
6.11. Các phép toán đạo hàm	110
6.11.1. Các đạo hàm bậc nhất	111
6.11.2. Các đạo hàm bậc hai	116
6.11.3. Các bộ lọc khác	121
6.12. Các phép toán morphology	122
6.12.1. Các định nghĩa cơ bản	124
6.12.2. Phép phát triển và phép bào mòn	124
6.12.3. Phép chập logic	129
6.12.4. Phép mở và phép đóng	131
6.12.5. Phép trùng – trật	132
6.12.6. Tóm tắt những phép toán cơ bản	133
6.12.7. Phép rút xương	134
6.12.8. Phép lan truyền	138
6.12.9. Tóm tắt về phép rút xương & phép lan truyền ..	139
6.12.10. Xử lý giá trị xám theo kiểu morphology	141
6.12.11. Làm trơn theo kiểu morphology	144
6.12.12. Tính gradient theo kiểu morphology	144
6.12.13. Toán tử Laplace theo kiểu morphology	145
6.12.14. Tóm tắt về các bộ lọc morphology	145

CHƯƠNG 7: MỘT SỐ KỸ THUẬT HIỆU CHỈNH VÀ PHÂN ĐOẠN ẢNH	147
7.1. Kỹ thuật hiệu chỉnh sự tạo bóng	147
7.1.1. Mô hình tạo bóng	148
7.1.2. Ước lượng sự tạo bóng	148
7.2. Các kỹ thuật phục hồi và tăng cường ảnh cơ bản ..	153
7.2.1. Kỹ thuật tạo mặt nạ không sắc nét	154
7.2.2. Khử nhiễu	155
7.2.3. Khử độ méo của ảnh	157
7.3. Phân đoạn ảnh	160
7.3.1. Kỹ thuật lấy ngưỡng	160
7.3.2. Kỹ thuật dò cạnh	166
7.3.3. Kỹ thuật morphology nhị phân	173
7.3.4. Kỹ thuật morphology mức xám	179
TÀI LIỆU THAM KHẢO	183

Chương 1

KHÁI QUÁT VỀ XỬ LÝ ẢNH

1.1. GIỚI THIỆU CHUNG

Nhờ có công nghệ số hóa hiện đại, ngày nay con người đã có thể xử lý tín hiệu nhiều chiều thông qua nhiều hệ thống khác nhau, từ những mạch số đơn giản cho đến những máy tính song song cao cấp. Mục tiêu của việc xử lý này có thể chia làm ba hướng như sau:

- ◆ Xử lý ảnh ban đầu để có được ảnh mới theo một yêu cầu xác định (Ví dụ như ảnh mờ cần xử lý để được ảnh rõ hơn).
- ◆ Phân tích ảnh để thu được các thông tin đặc trưng giúp cho việc phân loại, nhận biết ảnh (Ví dụ: phân tích ảnh vân tay để trích chọn đặc trưng vân tay).
- ◆ Hiệu ảnh đầu vào để có những mô tả về ảnh ở mức cao hơn, sâu hơn (Ví dụ: từ ảnh một tai nạn giao thông phác họa hiện trường tai nạn).

Trong giáo trình này, chúng ta sẽ tập trung vào những khái niệm cơ bản của *xử lý ảnh* và giới hạn vấn đề trong phạm vi ảnh 2-chiều, mặc dù rằng nhiều khái niệm và kỹ thuật được trình bày ở đây hoàn toàn có thể mở rộng cho các ảnh số chiều cao hơn.

Chúng ta bắt đầu với những định nghĩa cơ bản nhất trong xử lý ảnh. Một ảnh trong thế giới được xem là một hàm

hai biến thực $a(x,y)$ với a là độ sáng của ảnh tại vị trí tọa độ thực (x,y) . Một ảnh còn có thể chứa những ảnh con gọi là các vùng quan tâm, hay viết tắt theo tiếng Anh là ROI (Region Of Interest). Các ROI này vẫn thường được gọi tắt là các vùng. Khái niệm vùng phân ánh một thực tế là trong ảnh thường chứa nhiều đối tượng, mỗi đối tượng tạo nên phần cơ sở của một vùng. Đối với một hệ xử lý ảnh cao cấp, chúng ta có thể áp dụng nhiều phép toán cho từng vùng ảnh một, ví dụ như một vùng ảnh này sẽ được áp dụng các phép toán bỏ hiệu ứng mờ do chuyển động, trong khi một vùng ảnh khác sẽ được xử lý để nâng cao chất lượng màu sắc của nó.

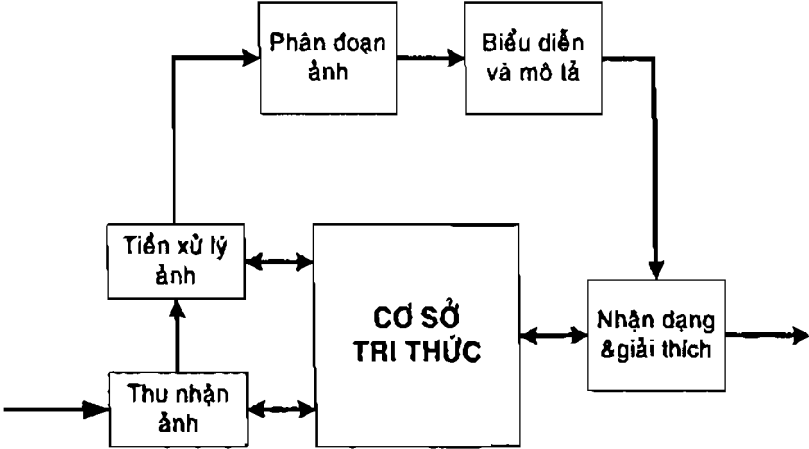
Các giá trị độ sáng của ảnh thường được thể hiện dưới dạng số thực hoặc số nguyên. Thông thường, những giá trị sáng kiểu số nguyên là kết quả của một quá trình lượng hóa chuyển một thang đo liên tục (chẳng hạn như giữa 0 và 100%) thành một số mức rời rạc. Tuy nhiên, trong nhiều quá trình hình thành ảnh, độ lớn của tín hiệu là kết quả đếm số hạt photon ở từng thời điểm, do vậy độ lớn ấy dĩ nhiên đã được lượng hoá sẵn. Còn trong một số quá trình tạo ảnh khác, ví dụ như tạo ảnh siêu âm trong y khoa, phương pháp đo đặc vật lý trực tiếp sẽ cho ra các giá trị phức, mỗi giá trị phức này gồm một giá trị độ lớn kết hợp với một giá trị pha, và cả hai đều ở dạng số thực.

1.2. CÁC BƯỚC CƠ BẢN TRONG XỬ LÝ ẢNH

Quá trình xử lý một ảnh số đầu vào nhằm thu được một ảnh đầu ra mong muốn thường phải trải qua nhiều bước khác nhau. Hình 1 giới thiệu với chúng ta những bước cơ bản của quá trình đó.

Trong hình vẽ này, để hoàn thành được mục tiêu xử lý cụ thể ứng với mỗi bước, chúng ta sẽ phải sử dụng, đến rất

nhiều khái niệm, định nghĩa, công cụ, thuật toán, kỹ thuật, vốn đã được phát triển và cải tiến trong quá trình hình thành nên ngành xử lý ảnh như ngày nay. Những kiến thức ấy sẽ lần lượt được trình bày trong những chương tiếp theo của giáo trình này.



Hình 1. Các bước cơ bản trong xử lý ảnh

Để hiểu rõ hơn về các bước đã được nêu trong hình 1, chúng ta hãy xét một ví dụ ứng dụng xử lý ảnh cụ thể. Giả sử là chúng ta cần xây dựng một chương trình cho phép xử lý từng bì thư một và qua đó cho ra địa chỉ của từng cặp người gửi/ nhận dưới dạng các chuỗi ký tự.

Với bài toán vừa nêu, rõ ràng việc trước hết cần phải thực hiện là có được ảnh bề mặt chứa địa chỉ của từng bì thư. Trong xử lý ảnh, công việc này có tên gọi là bước thu nhận ảnh, còn trong sơ đồ của hình 1 nó chính là bước đầu tiên. Để thực hiện, chúng ta cần có một bộ cảm biến lấy ảnh và khả

năng số hóa những tín hiệu liên tục được sinh ra bởi bộ cảm biến đó. Bộ cảm biến ở đây có thể là một máy chụp ảnh đơn sắc/ màu, hoặc một máy chụp ảnh kiểu quét dòng cho ra một dòng ảnh ở một thời điểm cụ thể. Với máy chụp kiểu này, sự di chuyển của bì thư sẽ giúp tạo ra một ảnh 2-chiều ứng với bề mặt của nó. Trong trường hợp kết xuất cho máy chụp ảnh/ bộ cảm biến cung cấp chưa phải ở dạng số hóa, chúng ta còn phải cho kết xuất đó đi qua một bộ phận khác nữa gọi là *bộ chuyển đổi tương tự - số*, vẫn hay được gọi tắt là ADC (*Analog-to-Digital Converter*), để thu được dạng số hóa của ảnh. Mặc dù đây chỉ là công đoạn đầu tiên trong quá trình xử lý, kết quả của nó có thể ảnh hưởng rất nhiều đến những công đoạn kế tiếp tùy theo loại hình ứng dụng, chất lượng, và chủng loại của thiết bị lấy ảnh. Những kiến thức có liên quan đến bước xử lý này sẽ lần lượt được khảo sát chi tiết hơn nữa trong các chương 2, 3, 5 của giáo trình.

Sau khi đã có ảnh bề mặt của từng bì thư dưới dạng số hóa, công đoạn kế tiếp là **tiền xử lý** ảnh đó. Ở bước này, ảnh sẽ được cải thiện về độ tương phản, khử nhiễu, khử bóng, khử độ lệch,... với mục đích làm cho chất lượng ảnh trở nên tốt hơn nữa, chuẩn bị cho các bước xử lý phức tạp hơn về sau trong quá trình xử lý ảnh. Các chương 4, 5, 6 và các mục 7.1, 7.2 sẽ cung cấp cho chúng ta những kiến thức có liên quan đến công đoạn tiền xử lý ảnh.

Bước thứ ba trong quá trình xử lý ảnh thường là bước **phân đoạn ảnh**. Nói một cách dễ hiểu, phân đoạn ảnh có nghĩa là chia một ảnh đầu vào thành nhiều phần nhỏ khác nhau hay còn gọi là các đối tượng. Trông có vẻ đơn giản, nhưng thật ra việc phân đoạn tự động ảnh thành tập những đối tượng khác nhau là nhiệm vụ phức tạp nhất trong xử lý

ánh số hóa. Nếu kết quả phân đoạn ảnh chỉ dừng lại được ở mức thô thiển, thì toàn bộ những bước xử lý tiếp sau sẽ không cho kết quả tốt, và như vậy không thể đạt được thành công nào về mặt ứng dụng. Mặt khác, các thuật toán phân đoạn không đủ mạnh, hoạt động không ổn định cũng sẽ là nguồn gốc dẫn đến sự thất bại của một giải pháp xử lý ảnh. Trong bài toán đọc địa chỉ thư tín tự động đang xem xét, bước này đồng nghĩa với việc tách ra được từ nền ảnh những ký tự hay từ đã được biết trước là phần lớn thuộc về các “đối tượng địa chỉ”.

Kết quả của bước phân đoạn ảnh vừa được đề cập trên đây thường được cho dưới dạng dữ liệu điểm ảnh thô, trong đó hàm chứa biên của một vùng ảnh, hoặc tập hợp tất cả những điểm ảnh thuộc về chính vùng ảnh đó. Trong cả hai trường hợp, sự chuyển đổi dữ liệu thô này thành một dạng thích hợp hơn cho việc xử lý trong máy tính là hết sức cần thiết. Để chuyển đổi chúng, câu hỏi đầu tiên cần phải trả lời là nên biểu diễn một vùng ảnh dưới dạng biên hay dưới dạng một vùng hoàn chỉnh gồm tất cả những điểm ảnh thuộc về nó. Biểu diễn dạng biên cho một vùng phù hợp với những ứng dụng chỉ quan tâm chủ yếu đến các đặc trưng hình dạng bên ngoài của đối tượng, ví dụ như các góc cạnh và điểm uốn trên biên chẳng hạn. Biểu diễn dạng vùng lại thích hợp cho những ứng dụng khai thác các tính chất bên trong của đối tượng, ví dụ như vân ảnh hoặc cấu trúc xương của nó. Trong một số ứng dụng, cả hai cách biểu diễn biên và vùng đều cần thiết, ví dụ như trong bài toán đọc thư mà chúng ta đang quan tâm. Bài toán này là một ví dụ điển hình cho lớp những ứng dụng nhận dạng ký tự, vốn luôn cần đến các thuật toán làm việc trên hình dạng biên cũng như trên cấu trúc xương, cùng nhiều tính chất bên trong của đối tượng.

Sự chọn lựa cách biểu diễn thích hợp cho một vùng ảnh chỉ mới là một phần trong việc chuyển đổi từ dữ liệu ảnh thô sang một dạng thích hợp hơn cho các xử lý về sau. Chúng ta còn phải đưa ra một phương pháp mô tả dữ liệu đã được chuyển đổi đó sao cho những tính chất cần quan tâm đến sẽ được làm nổi bật lên, thuận tiện cho việc xử lý chúng. Trong xử lý ảnh, thuật ngữ **mô tả** hay **chọn đặc trưng** đề cập đến sự rút trích từ ảnh những đặc trưng cần thiết dẫn đến sự hình thành các thông tin định lượng này có thể giúp chúng ta phân biệt các lớp đối tượng khác nhau trong ảnh. Như trong bài toán đọc thư nói riêng và nhân dạng ký tự nói chung, các bộ mô tả như “hồ” (“lỗ”), “vịnh” là những đặc trưng rất mạnh giúp phân biệt các phần khác nhau của bảng ký tự cần nhận dạng.

Bước cuối cùng trong sơ đồ đã nêu ra ở hình 1 là **nhận dạng** và **giải thích**. Nhận dạng là công đoạn gán *nhãn* cho một đối tượng dựa trên thông tin do bộ mô tả của đối tượng đó cung cấp. Giải thích là công đoạn gán nghĩa cho một tập các đối tượng đã được nhận biết. Trong ví dụ bài toán đọc thư đang xét, việc nhận dạng ra một ký tự như chữ “c” chẳng hạn đòi hỏi sự liên kết giữa các bộ mô tả cho ký tự đó với nhãn “c”. Trong khi đó, việc giải thích thường là một bước khó hơn nữa so với nhận dạng, ví dụ như gán cho tập hợp gồm một dãy liên tiếp ba con số, theo sau là dấu “-“, rồi lại theo sau đó là dãy liên tiếp hai con số, dấu “-“, rồi lại đến một dãy liên tiếp bảy con số nữa, tức là chuỗi có dạng *xxx-xx-xxxxxxx*, một nghĩa là “*số điện thoại đầy đủ*”, bao gồm cả mã nước, mã vùng, và số điện thoại của từng nhà.

Cho đến đây chúng ta vẫn chưa đề cập gì đến sự cần thiết của *tri thức cho trước* hay mối tương tác giữa *cơ sở tri thức* và các bước xử lý, như có thể quan sát thấy trong sơ đồ ở

hình 1. Ở đây, tri thức về một không gian bài toán sẽ được mã hóa vào trong một hệ xử lý ảnh dưới dạng cơ sở dữ liệu tri thức. Tri thức được đề cập đến có thể chỉ đơn giản là sự chi tiết hóa các vùng trong ảnh, nơi được biết trước là sẽ có những thông tin đáng quan tâm để tìm ra lời giải cho bài toán, do vậy giúp hệ thống thu hẹp phạm vi tìm kiếm và dẫn đến việc tìm ra lời giải nhanh hơn. Trong bài toán đọc thư, các vùng đó chính là những vị trí trên bì thư nơi thường có sự xuất hiện của các dòng địa chỉ liên quan đến người gửi/ nhận, tức là phần trái trên và phải dưới của ảnh bì thư đã thu được. Tri thức biết trước ấy cũng có thể hết sức phức tạp, ví dụ như bao gồm một danh sách liệt kê tất cả các khiếm khuyết chính có thể xảy ra ứng với bài toán kiểm định vật liệu, hoặc cả một cơ sở dữ liệu ảnh chứa nhiều ảnh chụp từ vệ tinh có độ phân giải cao cho một vùng như trong những ứng dụng kiểu dò tìm sự thay đổi trên mặt đất (theo dõi sự hình thành và phát triển của một cơn bão, của sự xây dựng một tổ hợp quân sự, ...). Ngoài mục đích hướng dẫn cách thức làm việc phù hợp cho mỗi bước xử lý ảnh, cơ sở tri thức còn có thể giúp điều khiển mối tương tác giữa các bước xử lý với nhau. Trong hình 1, sự tương tác này được thể hiện thông qua các mũi tên có hai đầu thay vì các mũi tên một đầu dùng để liên kết bước xử lý này với một bước xử lý tiếp sau. Sự mô tả như trong sơ đồ chỉ ra rằng các bước xử lý thường liên lạc với nhau trên cơ sở tri thức biết trước về dạng của kết quả mà chúng cần phải tạo ra. Lấy ví dụ, để nhận biết được chuỗi gồm tổng cộng 13 con số với hai dấu “-” nêu ở trên là một số điện thoại đầy đủ, thì hệ thống đọc thư tự động cần được cài đặt với tri thức giúp nhận ra vị trí đáng tin cậy của chuỗi ký tự đó khi nó thật sự là một số điện thoại, trong mối quan hệ với những thành phần khác trong vùng địa chỉ của bì thư. Bằng cách sử dụng

tri thức biết trước như vậy, chúng ta còn có thể nhận được sự phản hồi từ các bước xử lý thông qua cơ sở tri thức, chẳng hạn như chuỗi đó ở đúng vị trí của một số điện thoại đầy đủ, nhưng thông tin phản hồi cho biết nó chỉ gồm có 14 thay vì 15 ký tự, từ đó kết luận được là có thể đã có hai ký tự dính nhau. Với tình huống này, sự phản hồi thông qua cơ sở tri thức sẽ giúp hệ thống ra quyết định cho bước phân đoạn ảnh tiếp tục thực hiện lại việc phân đoạn một cách chính xác hơn. Có thể xem đây là một ví dụ về sự tận dụng tri thức trong khi thực hiện các tác vụ xử lý ảnh.

Trong những bước xử lý ảnh được thảo luận trên đây, chúng ta vẫn chưa bàn đến bài toán hiển thị ảnh, cũng là một bài toán hết sức phức tạp. Nó sẽ được trình bày sau chương 5 của giáo trình. Tuy nhiên, có một điều quan trọng mà chúng ta cần phải lưu ý là trong sơ đồ ở hình 1, yêu cầu hiển thị ảnh có thể được đưa ra ở bất kỳ một bước xử lý nào. Hơn nữa, không phải ứng dụng xử lý ảnh nào cũng cần phải thực hiện hết tất cả các bước xử lý đã nêu, ví dụ như các ứng dụng chỉnh sửa ảnh nghệ thuật chỉ dừng lại ở bước tiền xử lý. Một cách tổng quát thì những chức năng xử lý bao gồm cả nhận dạng và giải thích thường chỉ có mặt trong các hệ thống phân tích ảnh tự động hoặc bán tự động, được dùng để rút trích ra những thông tin quan trọng từ ảnh, ví dụ như các ứng dụng nhận dạng ký tự quang học.

1.3. TÀI NGUYÊN HỌC TẬP

Xử lý ảnh là một ngành nghiên cứu trẻ rất năng động trong những thập niên gần đây, do vậy có một khối lượng khá lớn các công trình nghiên cứu lý thuyết và ứng dụng đã được xuất bản. Phần lớn các công trình này có thể tìm thấy trên các

web site nghiên cứu của từng nhà nghiên cứu một, cũng như trong các web site thư viện điện tử của IEEE (www.ieee.org), IEEE Computer Society (www.computer.org/publications/), ACM (www.acm.org), Elsevier Science (www.elsevier.nl). Sau đây là danh sách các tạp chí và sách tham khảo chuyên ngành phổ biến nhất về xử lý ảnh.

Tạp chí nghiên cứu chuyên ngành

IEEE Computer Graphics & Applications

IEEE Transactions on Multimedia

IEEE Transactions on Pattern Analysis and Machine Intelligence

Sách tham khảo kinh điển

Tựa đề	Tác giả	Nhà xuất bản
Digital Picture Processing	A. Rosenfeld and A.C.Kak	Academic Press, New York, 1976 (lần 1), 1981 (lần 2)
Digital Image Processing	W.K.Pratt	Wiley, New York, 1978.
Fundamental of Digital Image Processing	Anil K.Jain	Prentice–Hall, Englewood Cliffs, New Jersey, 1989.
Rafael C.Gonzalez, Richard E.Woods	Digital Image Processing	Addison – Wesley, 1993

Bạn đọc cũng có thể tìm thấy rất nhiều nguồn tư liệu khác trên Internet dựa vào một số trang chủ chính, ví dụ như trang chủ sau đây: <http://www.ee.princeton.edu/~icip95/iplink/>. Ngoài ra, nếu bạn đọc muốn đi sâu vào nghiên cứu thì ứng với mỗi chủ đề được trình bày trong giáo trình này, bao giờ cũng có kèm theo rất nhiều các bài báo, công trình có liên quan được liệt kê trong phần Tài liệu tham khảo ở cuối sách.

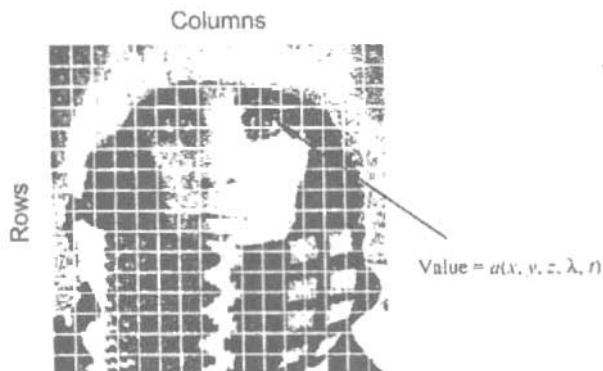
Chương 2

ẢNH SỐ HÓA VÀ SỰ CẢM NHẬN THỊ GIÁC

Một ảnh số $a[m,n]$ được mô tả trong không gian rời rạc 2-chiều chính là kết quả của quá trình lấy mẫu ảnh ở dạng tương tự $a(x,y)$ trong không gian liên tục 2-chiều. Quá trình này vẫn hay được đề cập đến với tên gọi là quá trình số hóa, với tác động của nó trên một ảnh liên tục được chỉ ra trong hình 2, còn bản chất toán học được khảo sát trong chương 3 của giáo trình.

Ảnh liên tục 2-chiều $a(x,y)$ được chia thành N hàng và M cột. Phần giao của mỗi hàng và cột được gọi là một *pixel* (*điểm ảnh*). Các giá trị độ sáng được gán cho các tọa độ nguyên $[m,n]$, với $\{m = 0, 1, 2, \dots, M-1\}$ và $\{n = 0, 1, 2, \dots, N-1\}$, tạo thành một mảng $a[m,n]$. Trong thực tế thì gần như ở tất cả các trường hợp, $a(x, y)$ – vốn có thể xem là một tín hiệu vật lý tiếp xúc với bề mặt của một bộ cảm biến 2-chiều – thật ra là một hàm nhiều biến, gồm các biến như **độ sâu (z)**, **màu sắc (λ)**, **thời gian (t)**. Trong chương này chúng ta chỉ xét các ảnh tĩnh đơn sắc 2-chiều.

Ảnh trong hình 2 đã được chia thành $N = 16$ hàng và $M = 16$ cột. Giá trị được gán cho mỗi pixel chính là giá trị độ sáng trung bình của nó được làm tròn thành số nguyên gần nhất. Quá trình biểu diễn độ lớn của một tín hiệu 2-chiều tại một tọa độ cụ thể bằng một con số nguyên với L mức xám khác nhau thường được gọi là *quá trình lượng hóa*.



Hình 2. Quá trình số hóa một ảnh liên tục. Pixel ở tọa độ $[m = 10, n = 3]$ có giá trị độ sáng là số nguyên bằng 110.

2.1. CÁC GIÁ TRỊ THÔNG DỤNG

Bảng 1 dưới đây liệt kê một số giá trị chuẩn cho ba tham số thông dụng nhất trong xử lý ảnh là hàng, cột, và các mức xám.

Bảng 1. Các giá trị phổ biến trong xử lý ảnh

Tham số	Ký hiệu	Các giá trị chuẩn
Hàng	N	256, 512, 525, 625, 1024, 1035
Cột	M	256, 512, 768, 1024, 1320
Các mức xám	L	2,64, 256, 1024, 4096, 16384

Chúng ta rất hay gặp trường hợp $M = N = 2^k$ với ($K = 8, 9, 10$) khi thực hiện một số thuật toán như biến đổi Fourier nhanh (xem mục 6.3 của Chương 6).

Bạn đọc cũng nên lưu ý là số các mức xám phân biệt thường ở dạng lũy thừa của 2, tức là $L = 2^B$, trong đó B là số bit dùng trong biểu diễn nhị phân của các giá trị độ sáng

ảnh. Khi $B > 1$ chúng ta có *ảnh mức xám*, còn khi $B = 1$ chúng ta có *ảnh nhị phân*. Ảnh kiểu này chỉ có hai mức xám là “đen” và “trắng”, hay “0” và “1”.

2.2 NHỮNG ĐẶC TRƯNG CỦA CÁC PHÉP TOÁN TRÊN ẢNH

Ứng với mỗi lớp ghép toán trên ảnh cho trước, người ta thường có hai mối quan tâm. Mối quan tâm thứ nhất là lớp phép toán ấy cho ra kết quả cụ thể như thế nào trên ảnh. Mối quan tâm còn lại là phí tổn tính toán cần có để thực hiện những phép toán ấy. Muốn trả lời hai câu hỏi quan trọng này, chúng ta cần phải tiến hành phân loại các phép toán trên ảnh.

2.2.1. Kiểu phép toán trên ảnh

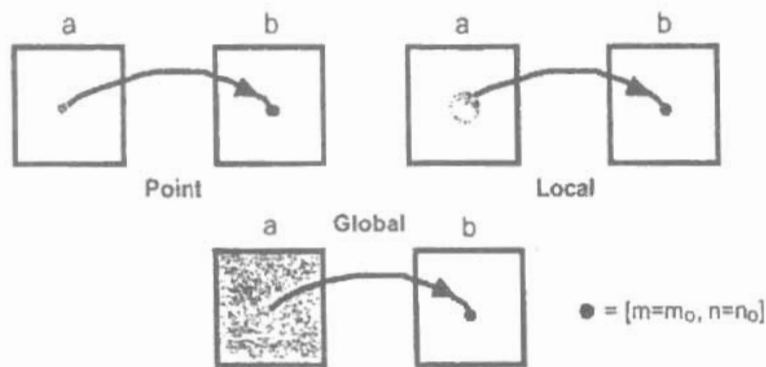
Một cách tổng quát, các phép toán trên ảnh cho phép biến đổi một ảnh đầu vào $a[m, n]$ thành một ảnh đầu ra $b[m, n]$ (hoặc thành một cách biểu diễn ảnh khác) có thể chia thành ba lớp như trong bảng 2 dưới đây.

Minh họa của các kiểu phép toán nói trên được cho trong hình sau đây.

Bảng 2. Các loại phép toán trên ảnh. Kích thước ảnh ở đây là $N \times N$, còn kích thước lân cận là $P \times P$. Chúng ta chú ý rằng độ phức tạp được tính bằng số phép toán ứng với mỗi pixel.

Phép toán	Tính năng	Độ phức tạp chung/ Pixel
<i>Trên pixel</i>	Giá trị đầu ra tại một tọa độ cụ thể chỉ phụ thuộc vào giá trị đầu vào tại cùng tọa độ đó.	<i>Hằng số</i>

<i>Cục bộ</i>	Giá trị đầu ra tại một tọa độ cụ thể phụ thuộc vào các giá trị đầu vào trong lân cận của tọa độ đó.	P^2
<i>Toàn cục</i>	Giá trị đầu ra tại một tọa độ cụ thể phụ thuộc vào tất cả các giá trị của ảnh đầu vào.	N^2



Hình 3. Minh họa các kiểu phép toán trên ảnh

2.2.2 Kiểu lân cận

Trong xử lý ảnh, những phép toán dựa trên lân cận giữ vai trò hết sức quan trọng. Chính vì lý do đó, chúng ta cần tìm hiểu cách thức ảnh được lấy mẫu, cũng như mối quan hệ giữa cách thức đó với rất nhiều kiểu lân cận có thể sử dụng trong quá trình xử lý một ảnh số.

Lấy mẫu theo hình chữ nhật

Trong hầu hết các trường hợp, ảnh sẽ được lấy mẫu bằng cách đặt lên trên nó một lưới chữ nhật, như đã chỉ ra trong

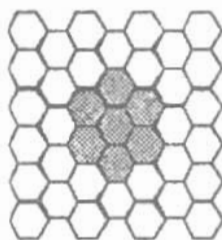
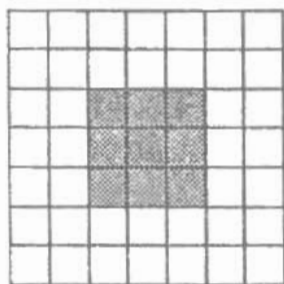
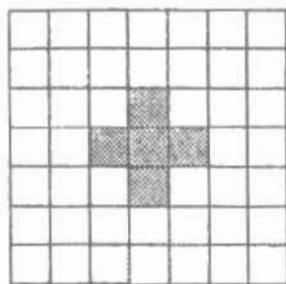
hình 2. Cách làm này dẫn đến kiểu lấy mẫu được nêu trong hình 4(a) và 4(b).

Lấy mẫu theo hình lục giác

Một cách lấy mẫu khác được minh họa trong hình 4(c). Nó được gọi một cách ngắn gọn là lấy mẫu lục giác.

Cả hai cách lấy mẫu nói trên đều đã được nghiên cứu khá nhiều [1] và điều có khả năng lặp đi lặp lại. Tuy nhiên do những ràng buộc về phần cứng và phần mềm, cách lấy mẫu theo hình chữ nhật được sử dụng phổ biến hơn.

Các phép toán cục bộ cho ra một giá trị kết xuất $b[m = m_0, n = n_0]$ dựa trên giá trị của những pixel nằm trong lân cận của pixel $b[m = m_0, n = n_0]$. Trong hình 4 dưới đây, chúng ta có thể thấy một số lân cận phổ biến, ví dụ như lân cận liên thông 4 và lân cận liên thông 8 (theo cách lấy mẫu hình chữ nhật), lân cận liên thông 6 (theo cách lấy mẫu lục giác).



Hình 4(a)

Lấy mẫu hình chữ nhật liên thông 4

Hình 4(b)

Lấy mẫu hình chữ nhật liên thông 8

Hình 4(c)

Lấy mẫu lục giác liên thông 6

2.3. CÁC THAM SỐ VIDEO

Do có nhiều ảnh tĩnh được thu nhận từ các máy quay video và các máy bắt khung, chúng ta cũng cần đề cập đến các tiêu chuẩn có liên quan đến ba dạng biểu diễn tín hiệu video phổ biến hiện nay trên thế giới là các hệ phát hình NTSC, PAL và SECAM. Thông tin về các chuẩn này được cho trong bảng 3 sau đây.

Bảng 3. Các tham số video chuẩn

Chuẩn	NTSC	PAL	SECAM
Số ảnh/giây	29.97	25	25
Lượng miligiây/ ảnh	33.37	40.0	40.0
Số dòng/ ảnh	525	625	625
(ngang/ dọc) = hệ số tỷ lệ	4:3	4:3	4:3
Kiểu đan xen	2:1	2:1	2:1

Trong một ảnh đan xen, các dòng số lẻ (1, 3, 5...) được quét ở nửa đầu của khoảng thời gian được cấp phát (tức là 20 ms trong hệ PAL), còn các dòng số chẵn được quét ở nửa sau (2, 4, 6...). Màn hình hiển thị ảnh khi đó cần phải được chia tọa độ theo định dạng quét này (xem mục 5.9.2). Nguyên nhân chính của việc tạo sự đan xen trong các dòng quét của một ảnh video là nhằm giảm độ rung của ảnh khi được hiển thị trên màn hình. Nếu chúng ta có ý định sử dụng các ảnh được lấy vào từ một nguồn video phát theo kiểu đan xen, điều quan trọng là chúng ta cần phải chú ý xem hai nửa ảnh có được “xê dịch” một cách phù hợp không thông qua phần cứng số hóa, hay là nó nên được cài đặt bằng phần mềm. Ngoài ra, việc

phân tích những đối tượng chuyển động đòi hỏi những chú ý đặc biệt đến nguồn video đan xen nhằm tránh được các cạnh chạy theo hình chữ chi.

Số hàng N của một nguồn phát video thường tương ứng một – một với số dòng của một ảnh video. Trong khi đó, số cột lại phụ thuộc vào cơ chế điện tử được dùng để số hóa ảnh. Các máy bắt khung khác nhau có thể sản sinh ra $M = 384$ hay 768 cột ứng với mỗi dòng cho cùng một nguồn video đầu vào.

2.4. SỰ CẢM NHẬN THỊ GIÁC

Ngày nay chúng ta có rất nhiều ứng dụng xử lý ảnh được thiết kế nhằm tạo ra các ảnh số phục vụ cho sự quan sát, thưởng thức của con người, thay vì phục vụ cho các quá trình giám định công nghiệp tự động như trong các nhà máy sản xuất có sự trợ giúp của máy tính. Khi thiết kế những ứng dụng như vậy, sự hiểu biết về các đặc trưng và giới hạn trong hệ thống thị giác của con người là hết sức quan trọng, vì chúng sẽ góp phần làm nên sự thành công của ứng dụng. Ít nhất chúng ta cũng phải ý thức được rằng:

- (1) Chưa có sự hiểu biết rõ ràng về hệ thống thị giác của nhân loại.
- (2) Hoàn toàn chưa có một độ đo nào có thể dùng để đánh giá chất lượng của hình ảnh dựa trên sự cảm nhận về chất lượng thị giác của con người.
- (3) Không tồn tại một mô hình quan sát kiểu mẫu nào cho hệ thống thị giác của con người.

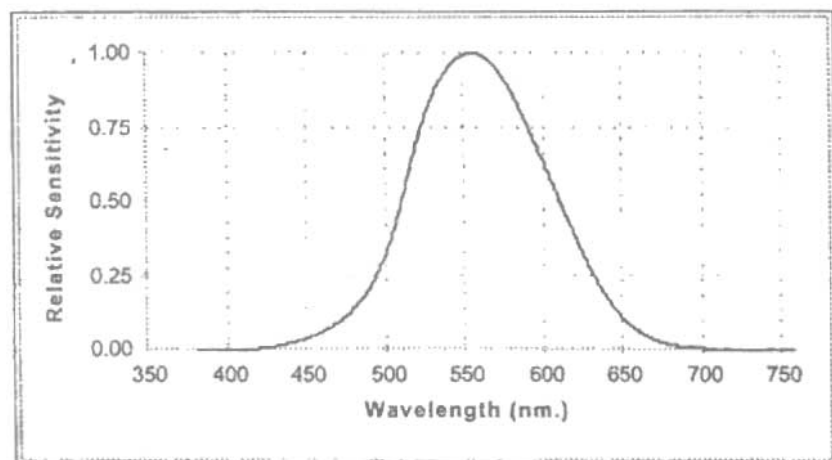
Mặc dù vậy, những nghiên cứu gần đây về tâm lý giác quan cũng đã mang lại cho con người những hiểu biết sâu sắc hơn về hệ thống thị giác của chính mình (xem Stockham [12]).

2.5. ĐỘ NHẠY SÁNG

Có rất nhiều cách mô tả độ nhạy trong hệ thống thị giác của con người. Chúng ta giả sử là một vùng đồng nhất trong một ảnh số sẽ có cường độ sáng cho bởi một hàm theo bước sóng (màu sắc) $I(\lambda)$. Ngoài ra $I(\lambda) = I_0$ là một giá trị không đổi.

2.5.1. Độ nhạy với bước sóng

Hình 5 dưới đây chỉ ra cường độ cảm nhận, hay còn gọi là độ nhạy phổ, dưới dạng một hàm theo bước sóng λ .



Hình 5. Độ nhạy phổ ứng với một bộ quan sát kiểu con người

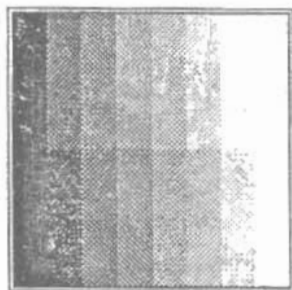
2.5.2. Độ nhạy với tác nhân kích thích

Nếu cường độ sáng không đổi I_0 được cho phép thay đổi thì với một xấp xỉ khá tốt, chúng có đáp ứng thị giác R sẽ tỷ lệ với logarit của cường độ đó. Kết quả này gọi là luật Weber-Fechner:

$$R = \log(I_0) \quad (1)$$

Những ý tưởng xoay quanh luật vật lý này rất dễ minh họa. Ở đây các bước cảm nhận bằng nhau về độ sáng $\Delta R = k$ đòi hỏi độ sáng vật lý (tác nhân kích thích) phải tăng theo lũy thừa. Điều này được chỉ ra trên hình 6(a) và 6(b).

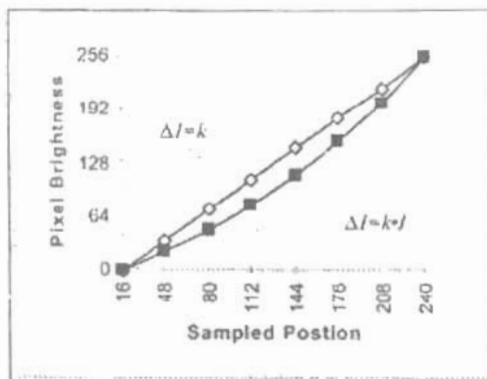
Một đường nằm ngang đi qua phần đỉnh của hình 6(a) chỉ ra có sự gia tăng tuyến tính về độ sáng khách quan (hình 6(b)), nhưng độ sáng chủ quan lại tăng theo hàm logarit. Một đường nằm ngang khác đi qua phần đáy của hình 6(a) thể hiện sự gia tăng lũy thừa trong độ sáng khách quan (hình 6(b)), nhưng độ sáng chủ quan lại tăng tuyến tính.



Hình 6(a)

(đỉnh) Bước sáng $-I = k$

(đáy) Bước sáng $-I = k.I$



Hình 6(b)

Các độ sáng thật sự cộng

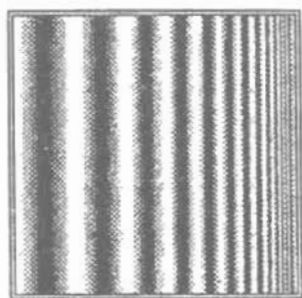
thêm các giá trị nội suy

Trong hình 6 (a) chúng ta có thể quan sát thấy hiệu ứng dải tần Mach. Mặc dù độ sáng vật lý là không đổi khi chúng ta đi qua các dải dọc, bộ quan sát của con người khi đó sẽ cảm nhận ra có một lượng thừa và thiếu về độ sáng tại vị trí cạnh của mỗi bước sáng. Do vậy ngay trước khi sang một bước sáng

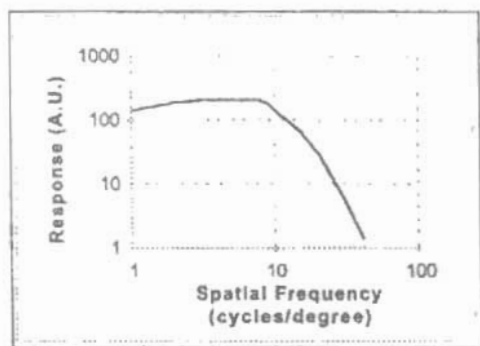
mới, chúng ta thấy có một sự giảm nhẹ về độ sáng so với giá trị vật lý thật sự. Còn ngay sau khi sang bước sáng mới, chúng ta lại thấy có một sự tăng nhẹ về độ sáng so với giá trị vật lý thật sự. Hiệu ứng tổng thể là sự gia tăng độ tương phản cảm nhận cục bộ tại vị trí cạnh của mỗi bước sáng.

2.6. ĐỘ NHẠY VỚI TẦN SỐ KHÔNG GIAN

Nếu cường độ sáng không đổi I_0 được thay thế bằng một sóng hình sin tăng dần về tần số không gian (hình 7(a)), chúng ta có thể đi xác định độ nhạy với tần số không gian như được chỉ ra trong hình 7 (b) [14, 15].



Hình 7(a). Sóng kiểm tra hình sin



Hình 7(b). Độ nhạy với tần số không gian

Để diễn dịch lại các dữ liệu này sang dạng thuật ngữ thông thường, chúng ta hãy xem một màn hình máy tính lý tưởng với khoảng cách quan sát là 50cm. Tần số không gian cho đáp ứng cực đại là 10 chu kỳ/ độ (xem hình 7(b)). Một độ quan sát ở mức 50cm sẽ được chuyển đổi thành $50 \times \tan(1^\circ) = 0.87\text{cm}$ trên màn hình máy tính. Do vậy tần số không gian

của đáp ứng cực đại $f_{\max} = 10 \text{ chu kỳ}/0.87 \text{ cm} = 11.46 \text{ chu kỳ/cm}$ tại khoảng cách quan sát này. Biến đổi công thức này thành dạng tổng quát hơn chúng ta có:

$$f_{\max} = \frac{10}{d \tan(1^\circ)} \text{ chu kỳ/cm} \quad (2)$$

trong đó d là khoảng cách quan sát được đo theo đơn vị cm.

2.7. ĐỘ NHẠY VỀ MÀU SẮC

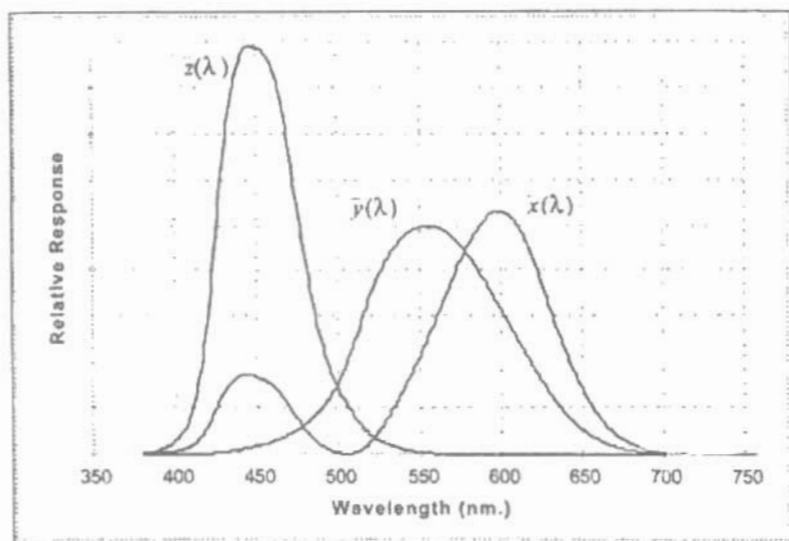
Sự cảm nhận của con người về màu sắc là một chủ đề vô cùng phức tạp. Do vậy trong mục này chúng ta chỉ trình bày tóm tắt một số kết quả quan trọng nhất. Như đã biết, sự cảm nhận màu sắc về mặt vật lý được dựa trên ba sắc màu có trong võng mạc của mắt người.

2.7.1 Bộ quan sát chuẩn

Dựa trên các phép đo tâm vật lý, Ủy ban CIE (Commission Internationale de l'Éclairage) đã thừa nhận một số đường cong chuẩn làm đường cong độ nhạy cho một bộ quan sát kiểu mẫu ứng với ba "sắc màu" $\bar{x}(\lambda)$, $\bar{y}(\lambda)$, $\bar{z}(\lambda)$, như được thể hiện trong hình 8. Đây không phải là các đặc trưng thu hút màu sắc thật sự trong một võng mạc chuẩn của con người, mà chỉ đơn thuần là các đường cong độ nhạy có được từ dữ liệu thực tế [10].

Trong một ánh, với mỗi vùng đồng nhất có cường độ sáng là hàm theo bước sóng (màu sắc) $I(\lambda)$, chúng ta có ba đáp ứng:

$$X = \int_0^{\tau} I(\lambda) \bar{x}(\lambda) d\lambda \quad Y = \int_0^{\infty} I(\lambda) \bar{y}(\lambda) d\lambda \quad Z = \int_0^{\infty} I(\lambda) \bar{z}(\lambda) d\lambda \quad (3)$$



Hình 8. Các đường cong độ nhạy màu ứng với một bộ quan sát chuẩn

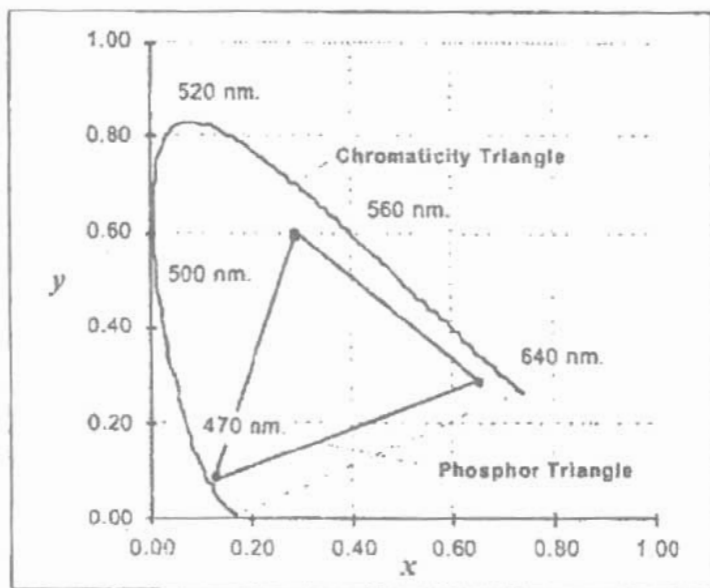
2.7.2. Các tọa độ màu CIE

Các tọa độ màu mô tả lượng thông tin màu sắc cảm nhận được bởi mắt người. Chúng được xác định thông qua các công thức sau đây:

$$x = \frac{X}{X+Y+Z} \quad y = \frac{Y}{X+Y+Z} \quad z = 1 - (x + y) \quad (4)$$

Trong các công thức trên, x đại diện cho tọa độ màu đỏ, còn y đại diện cho tọa độ màu lục. Các giá trị này tuyến tính trong $I(\lambda)$, do đó thông tin về cường độ tuyệt đối đã bị loại mất trong phép tính tọa độ màu (x,y) mà chúng ta vừa nêu. Toàn bộ các phân phối màu $I(\lambda)$, xuất hiện trước bộ quan sát với cùng một sắc màu sẽ có cùng tọa độ màu sắc theo cách tính này.

Nếu chúng ta sử dụng một nguồn màu thuần tinh chỉnh được, thì cường độ có thể được mô hình hóa dưới dạng $I(\lambda) = \delta(\lambda - \lambda_0)$ với $\delta(\bullet)$ là hàm xung. Tập hợp các tọa độ màu (x, y) sinh bởi sự biến thiên của λ_0 được gọi là tam giác màu CIE, như được minh họa trong hình 9.



Hình 9. Lược đồ màu chứa tam giác màu CIE liên kết với các màu có phổ thuần và tam giác ứng với các chất phát sáng trên màn hình CRT

Trong hình 9, các màu có phổ thuần sẽ nằm dọc theo biên của tam giác màu. Tất cả những màu còn lại nằm bên trong của tam giác. Bảng 4 dưới đây cho chúng ta tọa độ màu của một số nguồn sáng tiêu biểu.

Bảng 4. Tọa độ màu của một số nguồn sáng chuẩn

Nguồn	x	y
Đèn huỳnh quang với 4800° K	0.35	0.37
Mặt trời với 6000° K	0.32	0.33
Chất phát sáng đỏ (europium yttrium vanadate)	0.68	0.32
Chất phát sáng lục (zinc cadmium sulfide)	0.28	0.60
Chất phát sáng lam (zinc sulfide)	0.15	0.07

Sự mô tả màu sắc dựa trên các tọa độ màu không những cho phép phân tích màu còn giúp tổng hợp được chúng. Bằng cách trộn hai nguồn màu lại với nhau, chúng ta có thể tạo ra một màu bất kỳ nằm dọc theo đường thẳng nối các tọa độ màu tương ứng với hai màu đó. Do số lượng hạt photon không thể có giá trị âm nên các hệ số trộn phải dương. Bằng cách sử dụng ba nguồn màu như các chất phát sáng đỏ, lục, lam trên các màn hình CRT, chúng ta sẽ có được tập hợp màu được xác định bởi phần bên trong của “tam giác phát sáng” như trong hình 9.

Công thức để chuyển đổi từ các giá trị phát sáng (X, Y, Z) sang các màu màn hình CRT thông dụng (R, G, B) và ngược lại được cho trong các công thức sau đây:

$$\begin{bmatrix} R \\ G \\ B \end{bmatrix} = \begin{bmatrix} 1.9107 & -0.5326 & -0.2883 \\ -0.9843 & 1.9984 & -0.0283 \\ 0.0583 & -0.1185 & 0.8986 \end{bmatrix} \quad (5)$$

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} 0.6067 & 0.1736 & 0.2001 \\ 0.2988 & 0.5868 & 0.1143 \\ 0.0000 & 0.0661 & 1.1149 \end{bmatrix} \cdot \begin{bmatrix} R \\ G \\ B \end{bmatrix} \quad (6)$$

· Khi vị trí của một màu mong muốn (X, Y, Z) ở trong tam giác phát sáng của hình 9, các giá trị R, G, B tính được nhờ phương trình (5) luôn dương và do vậy có thể dùng để điều khiển sự phát màu trên màn hình CRT.

Ý tưởng cho rằng ứng với một độ dịch chuyển nhỏ trong lược đồ màu (hình 9), chúng ta sẽ thu được một cách tỷ lệ một lượng thay đổi nhỏ về màu sắc cảm nhận là không chính xác. Tính chất đó chỉ được xấp xỉ trong không gian màu có được từ thực nghiệm) – sau đây:

$$u' = \frac{4x}{-2x + 12y + 3} \quad v' = \frac{9y}{-2x + 12y + 3}$$

$$x = \frac{9u'}{6u' - 16v' + 12} \quad y = \frac{4v'}{6u' - 16v' + 12} \quad (7)$$

2.8. CÁC ẢO GIÁC QUANG HỌC

Sự mô tả về hệ thống thị giác của con người mà chúng ta vừa trình bày ở trên hoàn toàn được thể hiện thông qua các thuật ngữ kỹ thuật chuẩn mực. Điều này dẫn đến một kết luận là, chúng ta có đủ trí thức về hệ thống thị giác của con người để có thể mô hình hóa nó theo những kỹ thuật phân tích hệ thống tiêu biểu. Tuy nhiên, hai ví dụ đơn giản về ảo giác quang học mà chúng ta chỉ ra sau đây trên màn hình 10 cho thấy rằng cách tiếp cận có tính hệ thống như vậy đã đơn giản hóa vấn đề đi rất nhiều. Do vậy, khi sử dụng những mô hình đại loại như vậy, chúng ta phải hết sức cẩn thận.



Hình 10. Các ảo giác quang học

Ảnh bên trái trong hình 10 thể hiện ảo giác về các giá trị xám trong mắt người khi bộ não biết rằng thực tế chúng không tồn tại. Hơn thế nữa, chúng ta có cảm giác về một sự thay đổi động nào đó trong ảnh, mà một phần là do các chuyển động của mắt người. Ảnh bên phải trong hình 10, được gọi là tam giác Kanizsa, minh họa cho sự tăng cường về một tương phản và sự xuất hiện của các đường biên giả tạo [14]. Chúng ta sẽ không thể giải thích được những hiện tượng này dựa vào hệ thống cảm nhận thị giác mà chúng ta đã trình bày theo định hướng hệ thống ở trên.

Chương 3

LẤY MẪU ẢNH

Quá trình lấy mẫu ảnh là quá trình chuyển đổi một ảnh liên tục $a(x, y)$ sang dạng biểu diễn số hóa $b[m, n]$. Trong một hệ thống lấy mẫu lý tưởng, $a(x, y)$ sẽ được nhân với một tín hiệu huấn luyện xung lý tưởng 2-chiều:

$$\begin{aligned} b_{\text{ideal}}[m, n] &= a(x, y) \cdot \sum_{m=-\infty}^{+\infty} \sum_{n=-\infty}^{+\infty} \delta(x - mX_0, y - nY_0) \\ &= \sum_{m=-\infty}^{+\infty} \sum_{n=-\infty}^{+\infty} a(mX_0, nY_0) \delta(x - mX_0, y - nY_0) \quad (8) \end{aligned}$$

với X_0 và Y_0 là các khoảng lấy mẫu, $\delta(x, y)$ là một hàm xung lý tưởng. Ở một điểm nào đó dĩ nhiên hàm xung $\delta(x, y)$ sẽ được chuyển đổi thành hàm xung rời rạc $\delta[m, n]$. Với cách lấy mẫu theo lưới vuông, chúng ta có $X_0 = Y_0$. Việc lấy mẫu thông qua một hàm xung tương ứng với sự lấy mẫu trên các điểm vô cùng nhỏ. Tuy nhiên cách lấy mẫu này lại không giống với tình huống thông thường trong hình 2. Để thể hiện tác động của một cửa chấp lấy mẫu hữu hạn $p(x, y)$, chúng ta có thể chỉnh sửa mô hình lấy mẫu lại như sau:

$$b[m, n] = (a(x, y) \otimes p(x, y)) \cdot \sum_{m=-\infty}^{+\infty} \sum_{n=-\infty}^{+\infty} \delta(x - mX_0, y - nY_0) \quad (9)$$

Tác động kết hợp của cửa chấp và sự lấy mẫu đã được hiểu rõ ràng nhất khi chúng ta xét biểu diễn của tín hiệu trên miền Fourier.

$$B[\Omega, \Psi] = \frac{1}{4\pi^2} \sum_{m=-\infty}^{+\infty} \sum_{n=-\infty}^{+\infty} A(\Omega - m\Omega_s, \Psi - n\Psi_s) \cdot P(\Omega - m\Omega_s, \Psi - n\Psi_s) \quad (10)$$

trong đó $\Omega_s = 2\pi/X_0$ là tần số lấy mẫu theo hướng x , $\Psi_s = 2\pi/Y_0$ là tần số lấy mẫu theo hướng y . Cửa chấp $p(x, y)$ thường có dạng vuông, tròn, hay Gauss với xác suất liên kết $P(\Omega, \Psi)$ (xem bảng 11). Bản chất tuần hoàn của phổ nói trên, như được mô tả trong phương trình (53), có thể được quan sát thấy rất rõ ràng từ phương trình (10).

3.1. ĐỘ TRÙ MẬT LẤY MẪU TRONG XỬ LÝ ẢNH

Để ngăn ngừa hiện tượng *aliasing* (chồng lấp lên nhau) của các số hạng phổ như có thể thấy trước được từ phương trình (10), hai điều kiện sau cần phải được thỏa mãn:

A(u,v) là tín hiệu có dải tần giới hạn -

$$|A(u,v)| \equiv 0 \quad \text{với } |u| > u_c \text{ và } |v| > v_c \quad (11)$$

Tần số lấy mẫu Nyquist -

$$\Omega_s > 2 \cdot u_c \quad \text{và} \quad \Psi_s > 2 \cdot v_c \quad (12)$$

trong đó u_c và v_c là các tần số cắt theo hai hướng x và y tương ứng. Các ảnh được thu nhận từ những thấu kính có tính đối xứng vòng, không có quang sai, có độ nhiễu xạ giới hạn nhìn chung sẽ có dải tần giới hạn. Các thấu kính đó hoạt động như một bộ lọc thông thấp với tần số cắt trong miền tần số phương trình (43) được cho bởi:

$$u_c = v_c = \frac{2NA}{\lambda} \quad (13)$$

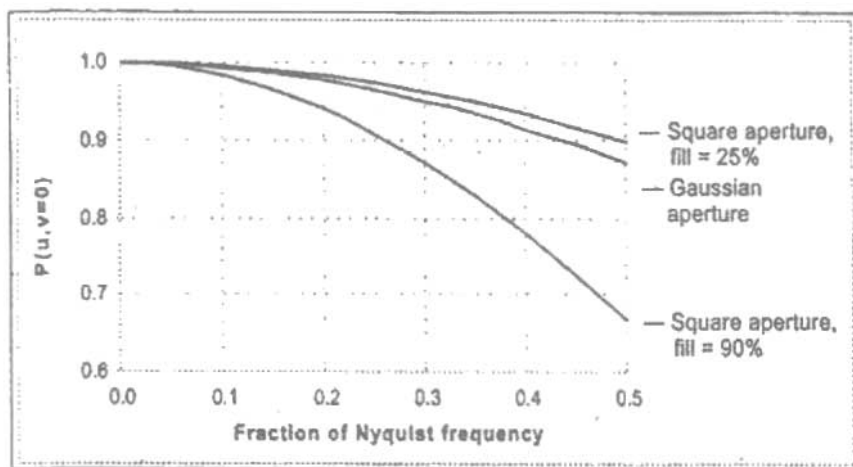
Trong đó NA là cửa chấp số học của các thấu kính, λ là độ dài bước sóng ngắn nhất được sử dụng cùng với những thấu kính đó [16]. Nếu các thấu kính không đạt được tiêu chuẩn nào trong ba tiêu chuẩn đã giả định trước, nó vẫn sẽ có dải tần giới hạn nhưng có các tần số cắt thấp hơn so với những tần số đã cho trước trong phương trình (13). Khi làm việc với tần số F quang học thay vì với NA trong không khí (nơi có chỉ số khúc xạ là 1.0), phương trình (13) trở thành:

$$u_c = v_c = \frac{2}{\lambda} \left(\frac{1}{\sqrt{4F^2 + 1}} \right) \quad (14)$$

3.1.1. Cửa chấp lấy mẫu

Cửa chấp $p(x,y)$ được mô tả ở trên chỉ gây ra một hiệu ứng nhỏ trên tín hiệu sau cùng nếu các điều kiện trong hai phương trình (12) và (13) được thỏa mãn. Giả sử như khoảng cách giữa các mẫu X_0 bằng Y_0 và cửa chấp lấy mẫu không rộng hơn X_0 , thì hiệu ứng xảy ra trên toàn phổ (do ứng xử của đại lượng $A(u,v)P(u,v)$ gây ra theo phương trình (9)) sẽ có dạng như trong hình 11 cho các cửa chấp Gauss và cửa chấp vuông.

Trong hình 11, phổ tín hiệu được ước lượng dọc theo một trục của phép biến đổi Fourier 2-chiều. Cửa chấp Gauss trong hình này có bề rộng sao cho khoảng lấy mẫu X_0 chứa $\pm 3\sigma$ (99.7%) của phân số Gauss. Các cửa chấp chữ nhật có bề rộng thỏa một trong số chúng chiếm 95% khoảng lấy mẫu, trong khi các cửa chấp còn lại chiếm 50% khoảng lấy mẫu. Độ rộng 95% này sẽ được chuyển đổi thành một hệ số làm đầy 90%, còn độ rộng 50% thành hệ số làm đầy 25%. Hệ số làm đầy được giới thiệu trong mục 5.5.2.



Hình 11. Phổ cửa chấp $P(u, v = 0)$ cho các tần số bằng $1/2$ tần số Nyquist

3.2. ĐỘ TRÙ MẬT LẤY MẪU TRONG PHÂN TÍCH ẢNH

Các nguyên tắc chọn độ trù mật lấy mẫu trong phân tích ảnh có nhiều điểm khác biệt so với trong xử lý ảnh. Điểm khác biệt cơ bản nhất ở đây là sự số hóa các đối tượng trong ảnh thành một tập các pixel sẽ dẫn đến một dạng nhiễu lượng hóa không gian có dải tần không giới hạn. Sự xuất hiện của loại nhiễu này dẫn đến những kết quả sau đây trong việc chọn lựa độ trù mật lấy mẫu nhằm phục vụ cho việc đo diện tích và tính độ dài chu vi của một vùng hay đối tượng.

3.2.1. Lấy mẫu để đo diện tích

Chúng ta giả sử rằng cách lấy mẫu được tiến hành theo kiểu lấy mẫu vuông, tức là $X_0 = Y_0$, và thuật toán ước lượng diện tích là không thiên vị theo kiểu đếm số pixel. Khi đó giá trị CV (xem phương trình (70)) của phép đo diện tích có liên hệ như sau với độ trù mật lấy mẫu [17]

$$2D: \lim_{S \rightarrow \infty} CV(S) = k_2 S^{-3/2} \quad 3D: \lim_{x \rightarrow \infty} CV(S) = k_3 S^{-2} \quad (15)$$

Còn trong không gian D-chiều:

$$\lim_{S \rightarrow \infty} CV(S) = k_D S^{-(D+1)/2} \quad (16)$$

trong đó S là số mẫu trên mỗi đường kính đối tượng. Trong không gian 2-chiều giá trị đo được là diện tích, trong không gian 3-chiều giá trị đó là thể tích, và trong không gian D-chiều nó là một siêu thể tích.

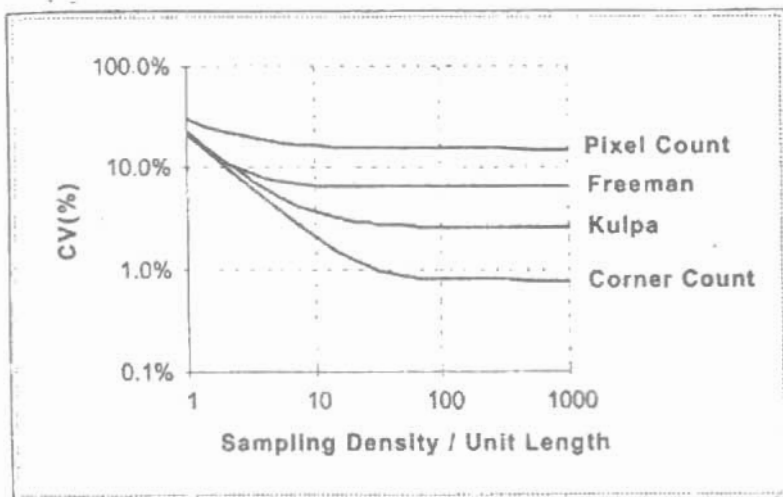
3.2.2. Lấy mẫu để đo độ dài

Một lần nữa chúng ta giả sử rằng cách lấy mẫu được tiến hành theo kiểu lấy mẫu vuông, tức là $X_0 = Y_0$, và các thuật toán ước lượng độ dài được dựa trên biểu diễn mã xích của Freeman (xem mục 6.6.1). Khi đó giá trị CV của phép đo độ dài sẽ có liên hệ với độ trù mật lấy mẫu (tính theo đơn vị độ dài) theo như trong hình 12 (xem [18] [19]).

Các đường cong trong hình 12 được phát triển trong ngữ cảnh là những đường thẳng, tuy nhiên chúng ta cũng có những kết quả tương tự cho đường cong và các đường biên đóng. Công thức ước lượng độ dài trong trường hợp này sử dụng biểu diễn mã xích của một đường thẳng và có sự tham gia của ba tham số theo như phương trình dưới đây.

$$L = \alpha \cdot N_e + \beta \cdot N_0 + \gamma \cdot N_c \quad (17)$$

với N_e là các mã xích chẵn, N_0 là số các mã xích lẻ, còn N_c là số các góc. Các công thức cụ thể được cho trong bảng 5.



Hình 12. Giá trị CV của phép đo độ dài ứng với nhiều thuật toán khác nhau

3.2.3. Kết luận về lấy mẫu

Trong trường hợp ứng dụng cần xây dựng chủ yếu có liên quan đến xử lý ảnh, chúng ta nên chọn một độ trù mật lấy mẫu tuân theo lý thuyết tín hiệu cổ điển dựa trên định lý lấy mẫu Nyquist. Trong trường hợp ứng dụng lại thiên về phân tích ảnh, chúng ta nên chọn độ trù mật lấy mẫu dựa trên các tiêu chí đo đặc không thiên vị (*bias*) và chính xác (CV). Trong trường hợp không thể xác định rõ là ứng dụng thiên về phía nào, có lẽ nên chọn độ trù mật có giá trị cao hơn trong số hai giá trị độ trù mật có thể có được.

Bảng 5. Các công thức ước lượng độ dài dựa trên các tham số có liên quan đến mã xích (N_e, N_o, N_d)

Các hệ số	α	β	γ	
Công thức				Tài liệu tham khảo
Đếm số pixel	1	1	0	[18]
Freeman	1	$\sqrt{2}$	0	[11]
Kuipa	0.9481	$0.9481 * \sqrt{2}$	0	[20]
Đếm số góc	0.980	1.406	-0.091	[21]

Chương 4

CÁC LOẠI NHIỄU

Ảnh số thu nhận được từ các bộ cảm biến hiện đại có thể bị suy giảm chất lượng do rất nhiều nguồn nhiễu khác nhau. Với khái niệm nhiễu chúng ta muốn đề cập đến những thay đổi ngẫu nhiên, thay vì những tác nhân làm lệch ảnh có tính tất định như hiệu ứng tạo bóng hay không đủ tiêu cự.

Trong chương này chúng ta sẽ khảo sát về nhiễu trên các ảnh được tạo nên từ các nguồn sáng trong những thiết bị quang điện tử hiện đại. Một cách cụ thể, thiết bị thu nhận ảnh là các máy chụp tích điện kép tiên tiến CCD (Charge-Coupled Device) với các photon sản sinh ra những điện tử thường được gọi là photoelectron. Mặc dù vậy, hầu hết các quan sát được trình bày ở đây về nhiễu, cũng như các nguyên nhân khác nhau của chúng đều có thể áp dụng được cho các mô hình lấy ảnh khác.

4.1. NHIỄU PHOTON

Khi tín hiệu được quan sát là ánh sáng, bản chất lượng tử của ánh sáng giữ vai trò hết sức quan trọng trong việc hình thành nhiễu. Một photon có bước sóng $\lambda = 500\text{nm}$ mang năng lượng $E = h\nu = hc/\lambda = 3.97 \times 10^{-19} \text{ jun}$. Với các photon như vậy, các máy chụp CCD hiện đại có đủ nhạy của máy chụp trong mục 5.2). Bài toán nhiễu xuất hiện ở đây do bản chất thống kê của quá trình sản sinh ra các photon. Chúng ta không thể giả định được rằng ứng với mỗi pixel cho trước,

trong hai thời khoảng quan sát liên tiếp nhưng độc lập có độ dài T , lượng photon đếm được là như nhau. Do quá trình tạo photon được điều khiển bởi các qui luật trong vật lý lượng tử, cách tốt nhất để khảo sát nhiễu là quan tâm đến lượng photon trung bình đếm được trong một cửa sổ quan sát cho trước. Phân phối xác suất cho p – photon trong một cửa sổ quan sát có độ dài T giây được biết là có dạng phân phối Poisson:

$$P(p|\rho, T) = \frac{(\rho T)^p e^{-\rho T}}{p!} \quad (18)$$

trong đó ρ là tham số tốc độ/ cường độ được đo bằng đơn vị số photon/giây. Chúng ta cần lưu ý rằng ngay cả khi không có nguồn nhiễu nào khác trong chuỗi ảnh, thì do các dao động thống kê trong việc đếm số lượng photon trên một khoảng hữu hạn T vẫn có một tỷ số tín hiệu – nhiễu hữu hạn SNR tồn tại. Bằng cách sử dụng công thức tính SNR thích hợp theo phương trình (29), và để ý là giá trị trung bình cùng với độ lệch chuẩn trong phân phối Poisson liên hệ với nhau theo công thức:

$$\text{Giá trị trung bình} = \rho T$$

$$\text{Tiến trình Poisson}$$

$$\sigma = \sqrt{\rho T} \quad (19)$$

Chúng ta sẽ có giá trị SNR được xác định bởi:

$$\text{Nhiều photon} - \text{SNR} = 10 \log_{10}(\rho T) \text{dB} \quad (20)$$

Với nhiễu photon, ba giả định truyền thống về mối quan hệ giữa tín hiệu và nhiễu sẽ không còn đúng nữa. Một cách cụ thể là:

- Nhiễu photon không độc lập với tín hiệu;

- Nhiều photon không có dạng Gauss;
- Nhiều photon không cộng tính.

Với những tín hiệu cực sáng giá trị – vượt quá 10^5 , các dao động nhiễu do tính thống kê photon gây ra có thể bỏ qua được nếu bộ cảm biến ảnh có mức bão hòa đủ cao. Điều này sẽ được bàn đến kỹ lưỡng hơn trong mục 5.3 và cụ thể là trong phương trình (29).

4.2. NHIỀU NHIỆT

Chúng ta cần để ý rằng nguồn điện tử ngẫu nhiên trong một giếng CCD là nguồn năng lượng nhiệt. Các điện tử có thể tự giải phóng chúng ta khỏi vật liệu CCD nhờ dao động nhiệt và sau đó bị bắt lại trong giếng CCD. Những điện tử như vậy rất khó phân biệt khi so với các photoelectron thật sự. Bằng cách làm mát chip CCD, chúng ta có thể giảm được đáng kể số lượng các “điện tử nhiệt” vốn là nguyên nhân dẫn đến nhiễu nhiệt hay hiện tượng dòng tối. Khi thời gian tích hợp T tăng lên, số lượng các điện tử nhiệt cũng sẽ tăng theo. Phân phối xác suất của chúng cũng là một tiến trình Poisson có tham số tốc độ là hàm tăng theo nhiệt độ. Có nhiều kỹ thuật làm mát khác nhau để loại bỏ bớt dòng tối, trong số đó thường bao gồm việc ước lượng *dòng tối trung bình* ứng với một khoảng thời gian tích hợp cho trước, sau đó loại thời gian này ra khỏi các giá trị pixel của CCD trước khi tiến hành chuyển đổi tương tự – số. Mặc dù phương pháp này giúp làm giảm giá trị trung bình của dòng tối, nó lại không thể giúp làm giảm độ lệch chuẩn của dòng tối, hơn thế nữa nó còn làm giảm thang độ sáng động của tín hiệu.

4.3. NHIỀU ĐIỆN TỬ TRÊN CHIP

Nhiều này có nguồn gốc trong quá trình đọc tín hiệu từ bộ cảm biến thông qua FET (Field Effect Transistor) của một chip CCD. Dạng tổng quát của độ trừ mật phổ năng lượng ứng với nhiều đọc ra là:

$$\text{Nhiều đọc ra} - S_{nn}(\omega) \propto \begin{cases} \omega^{-\beta} & \omega > \omega_{\max} & \beta > 0 \\ k & \omega_{\min} < \omega < \omega_{\max} \\ \omega^{\alpha} & \omega < \omega_{\min} & \beta > 0 \end{cases} \quad (21)$$

trong đó α và β là các hằng số, ω là tần số bán kính chuyển tín hiệu từ chip CCD ra thế giới bên ngoài. Ở các mức đọc ra cực thấp ($\omega < \omega_{\min}$) nhiều sẽ có tính chất $1/f$. Loại nhiều này có thể được làm giảm đến mức quản lý được nhờ sự chọn lựa các tốc độ đọc ra thích hợp nhất. Tuy nhiên ở mức tín hiệu cực thấp (xem phương trình (20)), nhiều đọc ra có thể vẫn trở thành một thành phần đáng kể trong giá trị SNR toàn thể [22].

4.4. NHIỀU KTC

Nhiều liên kết với điện trở cổng của một FET được gọi là nhiều KTC. Đây là nhiều hoàn toàn không thể bỏ qua được và có giá trị đầu ra RMS về điện thế được cho bởi công thức sau:

$$\text{Nhiều KTC (điện thế)} - \sigma_{\text{KTC}} = \sqrt{\frac{kT}{C}} \quad (22)$$

trong đó C là trở kháng chuyển cổng FET, k là hằng số Boltzmann, còn T là nhiệt độ tuyệt đối của chip CCD (được đo theo độ K). Bằng cách sử dụng các quan hệ $Q = C \cdot V = N_e \cdot e$

giá trị đầu ra RMS của nhiễu KTC tính theo số các photoelectron (N_e) sẽ được xác định bởi:

$$\text{Nhiều KTC (điện tử)} - \sigma_{N_e} = \frac{\sqrt{kTC}}{e^-} \quad (23)$$

trong đó e^- là điện tích nguyên tố. Với $C = 0.5\text{pF}$ và $T = 233\text{K}$ chúng ta có $N_e = 252$ điện tử. Giá trị này là nhiễu “một lần” trên mỗi pixel xuất hiện trong khoảng thời gian đọc tín hiệu ra, do vậy nó độc lập với khoảng thời gian tích hợp (xem mục 4.1 và 5.7). Các thiết kế điện tử thích hợp có sử dụng kỹ thuật lấy mẫu kép tương quan có thể triệt tiêu hoàn toàn nhiễu KTC này [22].

4.5. NHIỄU KHUẾCH ĐẠI

Nhiều loại này thường có tính chất cộng tính, có phân bố Gauss, và độc lập có tín hiệu. Trong những thiết bị điện tử được thiết kế tốt và hiện đại, nhiễu khuếch đại nhìn chung có thể bỏ qua được. Ngoại lệ phổ biến nhất trong số chúng là các máy chụp hình màu, với việc sử dụng khuếch đại trong kênh màu lục nhiều hơn ở kênh màu lam hay đỏ, từ đó dẫn đến có nhiễu nhiều hơn trong kênh màu lục (xem thêm mục 5.6).

4.6. NHIỄU LƯỢNG HÓA

Nhiều lượng hóa xuất hiện trong quá trình lượng hóa độ lớn tín hiệu và ở công đoạn chuyển đổi tương tự – số thông qua các bộ ADC. Nó có tính chất cộng tính và độc lập với tín hiệu khi số các mức lượng hóa $L \geq 16$. Điều này tương đương với $B - 4$ bit (xem mục 2.1). Với một tín hiệu đã được chuyển đổi thành dạng điện tử và do vậy có các giá trị lớn nhất cũng như nhỏ nhất, phương trình (72) là công thức phù hợp nhất

cho việc xác định giá trị SNR. Nếu bộ chuyển đổi ADC được điều chỉnh sao cho 0 tương ứng với giá trị điện tử nhỏ nhất, còn $2^B - 1$ tương ứng với giá trị điện tử lớn nhất thì:

$$\text{Nhiều lượng hóa} - \text{SNR} = 6B + 11\text{dB} \quad (24)$$

Với $B \geq 8\text{bit}$, điều này có nghĩa là $\text{SNR} \geq 59\text{dB}$. Nhiều loại này có thể bỏ qua được khi giá trị SNR tổng thể của một hệ thống hoàn toàn bị thống trị bởi một loại nhiễu cho SNR *nhỏ nhất*. Trong các máy chụp CCD đó chính là nhiễu photon.

Chương 5

CÁC THIẾT BỊ THU ẢNH VÀ HIỂN THỊ ẢNH

Các thiết bị thu nhận ảnh ngày nay bao gồm rất nhiều chủng loại và được chế tạo dựa trên rất nhiều công nghệ khác nhau, từ đơn giản đến phức tạp tùy theo mục đích sử dụng. Chúng ta không có tham vọng trình bày hết về các thiết bị thu ảnh trong chương này, mà chỉ giới hạn trong việc khảo sát các tính năng kỹ thuật có liên quan đến loại máy chụp phổ biến nhất được dùng trong xử lý ảnh là máy chụp CCD.

5.1. TÍNH TUYẾN TÍNH

Mối quan hệ thường được mong muốn giữa một tín hiệu vật lý đầu vào (các photon) với một tín hiệu đầu ra (điện thế) là mối quan hệ tuyến tính. Một cách hình thức điều này có nghĩa là (như trong phương trình (80) nếu chúng ta có hai ảnh a và b , cùng với hai hằng số phức tùy ý w_1 và w_2 , thêm vào đó là một đáp ứng camera tuyến tính thì:

$$c = \mathcal{R}\{w_1 a + w_2 b\} = w_1 \mathcal{R}\{a\} + w_2 \mathcal{R}\{b\} \quad (25)$$

trong đó $\mathcal{R}\{\bullet\}$ là đáp ứng của camera, c là kết xuất của camera đó. Trong thực tế mối quan hệ giữa ảnh đầu vào a và ảnh đầu ra c thường được xác định bởi:

$$c = gain \bullet a^\gamma + offset \quad (26)$$

Ở đây γ là giá trị gamma của thiết bị ghi ảnh. Với một hệ ghi ảnh tuyến tính thật sự chúng ta phải có $\gamma = 1$ và $offset = 0$, nhưng do giá trị offset trong thực tế gần như không bao giờ

bằng 0, chúng ta phải chú ý bù trừ lại yếu tố này nếu như định rút trích ra các số đo về mật cường độ sáng. Các kỹ thuật bù trừ được khảo sát trong mục 7.1.

Bảng 6 sau đây liệt kê một số giá trị tiêu biểu của γ . Các camera hiện đại thường có khả năng chuyển đổi qua lại giữa các giá trị γ .

Bảng 6: So sánh các giá trị γ

Bộ cảm biến	Bề mặt	γ	Lợi điểm khi sử dụng
Chip CCD	Silicon	1.0	Tính tuyến tính
Ống vidicon	Sb ₂ S ₃	0.6	Nén thang độ sáng động – Các cảnh có độ tương phản cao
Phim	Silver halide	<1.0	Nén thang độ sáng động – Các cảnh có độ tương phản cao
Phim	Silver halide	> 1.0	Mở rộng thang độ sáng động – Các cảnh có độ tương phản thấp

5.2. ĐỘ NHẠY

Có hai cách để mô tả độ nhạy của một camera. Cách thứ nhất là đi xác định lượng nhỏ nhất các photoelectron có thể dò được qua đó thu được độ nhạy *tuyệt đối*. Cách thứ hai là mô tả số các photoelectron cần thiết để chuyển từ một mức sáng sang một mức sáng kế tiếp, hay còn được gọi là thay đổi một đơn vị tương tự – số (ADU). Cách làm này dẫn đến độ nhạy tương đối của camera.

5.2.1. Độ nhạy tuyệt đối

Để xác định độ nhạy tuyệt đối chúng ta cần đặc trưng hóa một camera dưới dạng nhiễu của nó. Nếu nhiễu tổng cộng

có chẳng hạn – 100 photoelectron, thì để bảo đảm tính dò được của một tín hiệu chúng ta có thể nói rằng ở mức 3σ , tín hiệu dò được tối thiểu (hay độ nhạy tuyệt đối) sẽ là 300 photoelectron. Nếu mọi nguồn nhiễu đã liệt kê trong chương 4, ngoại trừ nhiễu photon, đều có thể giảm đến mức bỏ qua được, chúng ta sẽ có được độ nhạy tuyệt đối không vượt quá 10 photoelectron với các công nghệ chế tạo CCD hiện đại.

5.2.2. Độ nhạy tương đối

Với định nghĩa đã nêu ở trên về độ nhạy tương đối S trong trường hợp tuyến tính (phương trình (26)) và $\gamma = 1$ chúng ta sẽ có:

$$S = \frac{1}{\text{gain}} = \text{gain}^{-1} \quad (27)$$

Quá trình đo *độ nhạy* hay *độ lợi* (*gain*) có thể thực hiện theo hai cách khác nhau sau đây.

Cách 1

Theo phương trình (26), nếu tín hiệu đầu vào a có thể được điều khiển chính xác thông qua thời gian tích hợp hoặc cường độ (nhờ các bộ lọc trừ mật trung tính), khi đó độ lợi có thể tính được qua hệ số góc của đường thẳng kết quả. Để chuyển đổi giá trị này sang các đơn vị đo mong muốn, chúng ta cần phải sử dụng một nguồn chuẩn phát ra một lượng photon biết trước lên bộ cảm biến camera, đồng thời độ hiệu quả lượng tử (η) cũng phải được cho trước. Ở đây độ hiệu quả lượng tử cho biết số photoelectron được sản sinh ra trên mỗi photon ứng với một độ dài sóng cho trước. Một cách tổng quát chúng ta có $0 \leq \eta(\lambda) \leq 1$.

Cách 2

Trong trường hợp hiệu ứng giới hạn của camera chỉ là nhiễu photon (xem mục 6.1), chúng ta còn có một kỹ thuật khác để cài đặt hơn để xác định được độ nhạy của camera. Bằng cách sử dụng phương trình (19), (26), (27) và sau khi bù trừ độ dịch chuyển offset (xem mục 7.1), độ nhạy đo được từ một cảnh c sẽ là:

$$S = \frac{E\{c\}}{\text{Var}\{c\}} = \frac{m_c}{s_c^2} \quad (28)$$

trong đó m_c và s_c được xác định thông qua các phương trình (66) và (68)

Bảng liệt kê các dữ liệu đo được ứng với năm kiểu camera CCD sản xuất năm 1995. Trong bảng này, với một camera CCD kiểu khoa học (C-1), chúng ta chỉ cần có 8 photoelectron (gần 16 photon) dùng để tách biệt hai mức xám trong biểu diễn số hóa của ảnh. Với một video camera ít tốn kém hơn (C-5), chúng ta cần phải có khoảng 256 photoelectron (gần 512 photon) để tách biệt hai mức xám. Như vậy độ nhạy của các camera biến đổi rất rõ rệt khi đi từ những sản phẩm thông thường cho đến chuyên dụng.

Bảng 7. Các giá trị độ nhạy ứng với nhiều loại camera CCD. Chúng ta cần chú ý rằng có nhiều hơn một camera có giá trị thấp hơn S

Camera Nhân	Pixel	Cỡ pixel $\mu\text{m} \times \mu\text{m}$	Nhiệt độ K	S e^-/ADU	Số bit
C-1	1320 x 1035	6.8 x 6.8	231	7.9	12
C-2	578 x 385	22.0 x 22.0	227	9.7	16

C-3	1320 x 1035	6.8 x 6.8	293	48.1	10
C-4	576 x 384	23.0 x 23.0	238	90.9	12
C-5	756 x 581	11.0 x 5.5	300	109.2	8

5.3. SNR

Như đã đề cập đến trong chương 4, trong các hệ thống camera hiện đại nhiều thường bị giới hạn bởi:

- Nhiễu khuếch đại trong trường hợp các camera màu.
- Nhiễu nhiệt. Nhiễu này tự giới hạn bởi nhiệt độ chip K và thời gian lộ sáng T, hoặc
- Nhiễu photon. Nhiễu này bị giới hạn bởi tốc độ sản sinh photon – và thời gian lộ sáng T.

5.3.1 Nhiễu nhiệt (dòng tối)

Bằng cách sử dụng những kỹ thuật làm mát dựa trên các phân tử làm mát Peltier, nhiệt độ chip có thể giảm xuống còn ở mức từ 230 đến 250 độ K, và khi đó tốc độ sản sinh ra các điện tử nhiệt sẽ giảm xuống. Để đo nhiễu nhiệt, chúng ta có thể chú ý đến khoảng thời gian cần thiết (tính theo giây) để tạo ra một lượng đủ lớn các điện tử nhiệt khi đi từ một mức sáng sang một mức sáng kế tiếp (một ADU) với sự thiếu vắng các photoelectron. Điều kiện cuối cùng này (sự thiếu vắng các photoelectron) chính là nguyên nhân dẫn đến tên gọi dòng tối. Các đặc trưng của nhiễu nhiệt ứng với năm loại camera tiêu biểu được cho trong bảng 8 dưới đây.

Bảng 8. Các đặc trưng của nhiễu nhiệt

Camera Nhãn	Nhiệt độ K	Đòng tối giây /ADU
C-1	231	526.3
C-2	227	0.2
C-3	293	8.3
C-4	238	2.4
C-5	300	23.3

Trong bảng trên, video camera (C-5) có sự giảm thiểu dòng tối trên chip (xem mục 4.2). Khi hoạt động ở nhiệt độ phòng camera này cần không nhiều hơn 20 giây để tạo ra một thay đổi ADU do nhiễu nhiệt. Điều này có nghĩa là với tốc độ phát hình từ 25 đến 30 ảnh/ giây (xem bảng 3) của camera, nhiễu nhiệt có thể bỏ qua được.

5.3.2. Nhiễu photon

Từ phương trình (20) chúng ta nhận thấy rằng nên tăng giá trị SNR bằng cách tăng thời gian tích hợp ảnh để “bắt” được nhiều photon hơn. Các pixel trong một camera CCD có dung lượng giếng hữu hạn C chính là giá trị SNR tối đa/pixel:

Nhiều photon có dung lượng giới hạn –

$$\text{SNR} = 10 \log_{10}(C) \text{ dB} \quad (29)$$

Bảng 9 cho chúng ta các dữ liệu có liên quan đến nhiễu photon đo được trong thực tế cũng như dựa trên lý thuyết. Chúng được thực hiện trên năm loại camera tiêu biểu.

Bảng 9. Các đặc trưng của nhiều photon

Camera Nhãn	C Số e	Giá trị SNR lý thuyết (dB)	Giá trị SNR đo được (dB)	Cỡ pixel $\mu\text{m} \times \mu\text{m}$	Độ sâu giếng Số $e^-/\mu\text{m}^2$
C-1	32000	45	45	6.8 x 6.8	692
C-2	340000	55	55	22.0 x 22.0	702
C-3	32000	45	43	6.8 x 6.8	692
C-4	400000	56	52	23.0 x 23.0	756
C-5	40000	46	43	11.0 x 5.5	661

5.4. TẠO BÓNG

Trong tất cả các hệ thống lấy ảnh chúng ta đều mong muốn thu được kết quả là với một ảnh vật lý đầu vào $a(x, y)$ không đổi thì phiên bản số hóa của ảnh cũng không đổi. Tuy nhiên trong thực tế khó lòng đạt được kết quả đó, do hầu hết các hệ lấy ảnh đều tạo ra bóng, với nguồn bóng có thể nằm ngoài camera do độ chiếu sáng trong cảnh lấy ảnh, hoặc do chính bản thân camera tạo nên khi giá trị *gain* và *offset* thay đổi từ pixel này sang pixel khác. Mô hình tạo bóng được xác định thông qua phương trình sau:

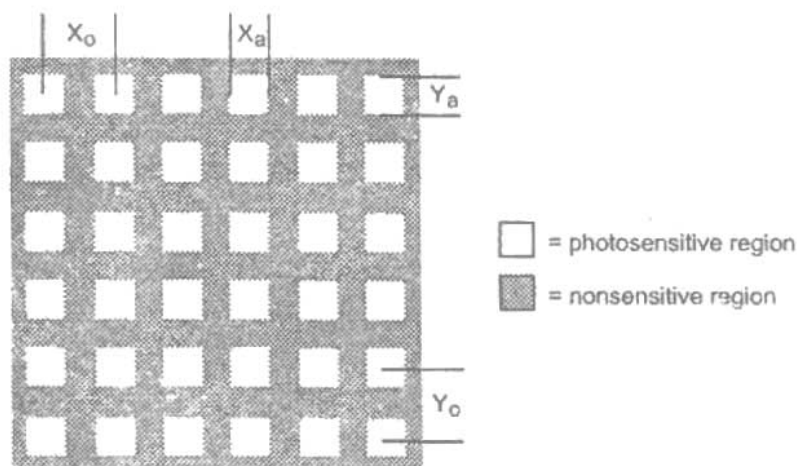
$$c[m, n] = \text{gain}[m, n] \cdot a[m, n] + \text{offset}[m, n] \quad (30)$$

Trong phương trình trên $a[m, n]$ là ảnh số hóa sẽ được ghi nhận nếu như không có hiện tượng tạo bóng trong ảnh, tức là $a[m, n] = \text{constant}$. Các kỹ thuật có liên quan đến việc làm giảm hoặc khử bóng có thể tìm thấy trong mục 7.1

5.5. DẠNG PIXEL

Mặc dù các pixel đã được định nghĩa trong hình 2 là luôn có dạng hình vuông và phủ toàn bộ ảnh liên tục, các tính

chất hình học của một hệ thống số hóa/ camera vẫn có những ảnh hưởng nhất định lên trên chúng. Hình 13 cho chúng ta định nghĩa của nhiều tham số có liên quan đến camera/ bộ số hóa và có ảnh hưởng đối với việc hình thành các pixel ảnh.



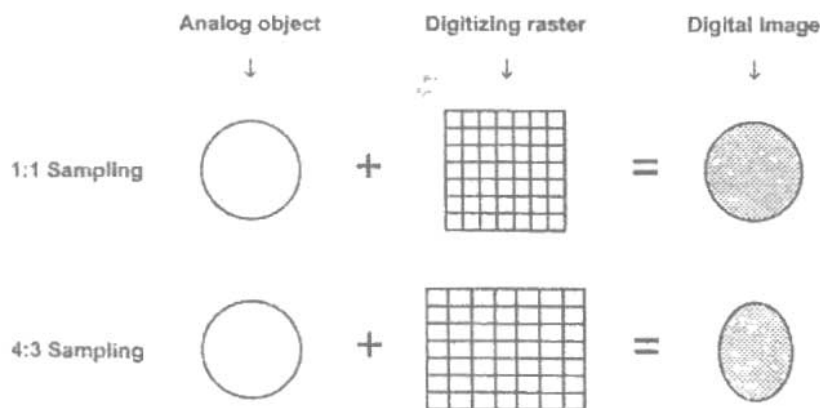
Hình 13. Các tham số dạng pixel

Trong hình 13, các tham số X_0 và Y_0 là khoảng cách giữa các tâm pixel. Chúng biểu thị các khoảng cách lấy mẫu theo phương trình (8). Các tham số X_a và Y_a là các kích thước của phần bề mặt camera nhạy với ánh sáng. Như đã đề cập trong mục 2.3, các bộ số hóa video khác nhau có thể có nhiều giá trị X_0 trong khi vẫn có chung một giá trị Y_0 .

5.5.1. Các pixel vuông

Như đã đề cập đến trong chương 3, việc lấy mẫu vuông sẽ dẫn đến $X_0 = Y_0$ hay $X_0/Y_0 = 1$. Trong thực tế các máy thu ảnh có $X_0/Y_0 = 1.1$ hoặc $X_0/Y_0 = 4/3$ cũng rất phổ biến, đặc

biệt là tỷ số 4/3 cho truyền hình thương mại. Độ rủi ro ứng với các pixel không vuông nằm ở chỗ khi các đối tượng đẳng hướng được quét vào thì trên màn hình camera chúng vẫn xuất hiện dưới dạng đẳng hướng, còn khi phân tích dựa trên các yếu tố có liên quan đến chiều dài và chiều rộng chúng sẽ cho ra các kết quả không đẳng hướng. Điều này được minh họa trên hình 14.



Hình 14. Tác động của các pixel không vuông

5.5.2. Hệ số làm đầy

Trong các camera CCD hiện đại có thể có tình huống một phần bề mặt của camera không nhạy sáng và được sử dụng để ngăn ngừa hiện tượng *blooming*. Hiện tượng *blooming* xuất hiện khi một giếng CCD được làm đầy (xem bảng 9) và các photoelectron phụ thêm tràn vào các giếng CCD kế cận. Các vùng anti-*blooming* giữa các site CCD thường được sử dụng để ngăn ngừa hiện tượng này xảy ra với giả định là sẽ có một phần nhỏ các photon đến bị mất đi khi

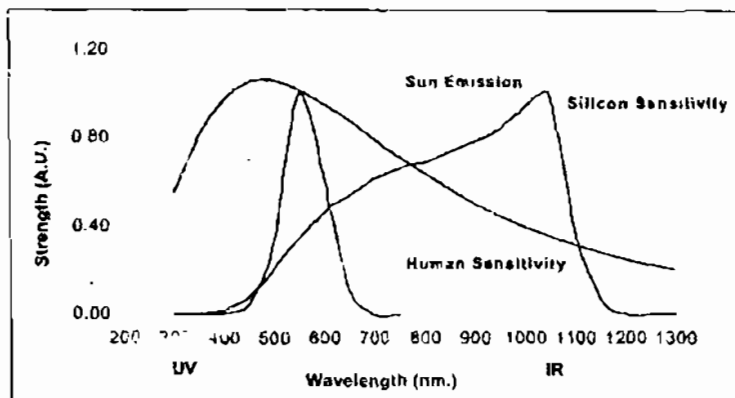
đập vào phân không nhạy sáng của chip CCD. Phần rất nhỏ bề mặt nhạy sáng được gọi là *hệ số làm đầy (fill-factor)* và được xác định theo công thức sau:

$$\text{fill - factor} = \frac{X_a \cdot Y_a}{X_0 \cdot Y_0} \times 100\% \quad (31)$$

Với giá trị *fill-factor* càng lớn, chip càng có khả năng bắt được nhiều ánh sáng cho đến mức tối đa là 100%, do vậy có thể giúp cải thiện giá trị SNR. Tuy nhiên giá trị của hệ số làm đầy càng lớn cũng dẫn đến độ trơ không gian nhiều hơn do hiệu ứng cửa chấp đã mô tả trong mục 3.1.1. Điều này được minh họa trong hình 17.

5.6. ĐỘ NHẠY PHỔ

Các bộ cảm biến được sử dụng trong camera và phim ảnh thường không có độ nhạy đồng nhất với tất cả các bước sóng, ánh sáng. Độ nhạy phổ cho một bộ cảm biến CCD được thể hiện trong hình 15.



Hình 15: Các đặc trưng phổ của silicon, mặt trời, và hệ thị giác của con người. UV viết tắt cho ultraviolet (tia cực tím), còn IR viết tắt cho infra-red (tia hồng ngoại).

Độ nhạy cao của silicon trong miền hồng ngoại cho thấy rằng, với các ứng dụng trong đó camera CCD được sử dụng như nguồn cung cấp ảnh cho quá trình phân tích và xử lý, việc sử dụng bộ lọc đồng khối IR cần phải được xem xét cẩn thận. Bộ lọc này sẽ đồng khối các bước sóng trên 750 nm, do vậy ngăn cản hiện tượng “sương mù” trong ảnh có được từ các bước sóng dài hơn trong ánh sáng mặt trời. Nói cách khác, một camera CCD có thể dùng để chế tạo nên một bộ cảm biến tuyệt hảo cho các bước sóng ở gần vùng hồng ngoại, trong khoảng từ 750nm đến 100nm.

5.7. TỐC ĐỘ CHỚP (THỜI GIAN TÍCH HỢP)

Khoảng thời gian để ánh sáng được phơi ra ánh sáng – hay nói cách khác là khoảng thời gian thu thập các photon – có thể thay đổi dựa trên chủng loại camera hoặc dựa trên định dạng video được sử dụng (xem bảng 3). Nó thường được gọi bằng một trong hai thuật ngữ là tốc độ chớp hoặc thời gian tích hợp.

5.7.1. Tốc độ chớp trên các video camera

Trên các video camera CCD kiểu thương mại tốc độ chớp thường ở mức 500 ns. Người ta còn sử dụng các tốc độ qui ước phổ biến là 33.37 ms cho hệ NTSC và 40.0 ms cho các hệ PAL & SECAM. Các giá trị tốc độ chớp cao cỡ 30 giây cũng có thể đạt được với một số camera nhất định, tuy nhiên làm như vậy chúng ta đã hy sinh tính liên tục của dòng dữ liệu ảnh video để có được một ảnh tích hợp duy nhất thay vì nhiều ảnh.

5.7.2. Tốc độ chớp trên các camera khoa học

Với các camera khoa học, tốc độ chớp có thể đạt mức 500ns. Với công nghệ làm mát chip dựa trên kỹ thuật làm

mát Peltier hoặc làm mát bằng nitơ lỏng, chúng ta hoàn toàn có thể đạt được thời gian tích hợp vượt quá một giờ đồng hồ trên các thiết bị chuyên dụng loại này.

5.8. TỐC ĐỘ ĐỌC RA

Tốc độ đọc dữ liệu từ chip cảm biến được gọi là tốc độ đọc ra. Tốc độ đọc ra cho các video camera chuẩn phụ thuộc vào nhiều tham số của bộ bắt ảnh cũng như chính bản thân camera. Với định dạng video chuẩn (xem mục 2.3), tốc độ đọc ra sẽ được cho bởi:

$$R = \left(\frac{\text{số ảnh}}{\text{giây}} \right) \left(\frac{\text{số dòng}}{\text{ảnh}} \right) \left(\frac{\text{số điểm ảnh}}{\text{dòng}} \right) \quad (32)$$

Mặc dù đơn vị mô tả thích hợp nhất cho tốc độ đọc ra là số pixel/ giây, người ta vẫn hay sử dụng thuật ngữ Hz để làm đơn vị biểu diễn. Với một camera sử dụng các pixel vuông (xem mục 5.5) chúng ta có bảng các tham số tương ứng sau.

Bảng 10. Các tốc độ đọc ra ứng với các định dạng video chuẩn

Định dạng	Số dòng/ giây	Số pixel/ dòng	R (MHz)
NTSC	15,750	$(4/3) * 525$	~ 11.0
PAL/ SECAM	15.625	$(4/3) * 625$	~ 13.0

Chúng ta cần chú ý rằng các số liệu trong bảng 10 chỉ có tính xấp xỉ. Giá trị chính xác cho một hệ thống sử dụng pixel vuông đòi hỏi tri thức chính xác về cách bộ số hóa video lấy mẫu mỗi dòng.

Các tốc độ đọc ra được sử dụng trong video camera hàm ý nhiều điện tử đã mô tả trong mục 6.3 xuất hiện trong vùng

phổ nhiều (phương trình (21)) mô tả bởi $\omega > \omega_{\max}$, là nơi năng lượng nhiều tăng theo tần số. Do vậy nhiều đọc ra có thể trở thành đáng kể trong các video camera.

Các camera khoa học thường sử dụng tốc độ đọc ra thấp hơn để làm giảm nhiễu đọc ra. Các tốc độ đọc ra cụ thể thường được sử dụng là 20 kHz, 500 kHz, 1 MHz cho đến 8 MHz.

5.9. CÁC THIẾT BỊ HIỂN THỊ ẢNH

Các màn hình được sử dụng trong xử lý ảnh – đặc biệt là các màn hình dùng chung với máy tính – có một số tính chất giúp xác định chất lượng của ảnh sau cùng.

5.9.1. Tốc độ sinh động

Tốc độ sinh động được định nghĩa là số các ảnh hoàn chỉnh ghi được ra màn hình trong mỗi giây. Với các định dạng video chuẩn, tốc độ này là cố định theo các giá trị đã cho trong bảng 3, tức là 29.97 hay 25 ảnh/ giây. Với màn hình máy tính, tốc độ sinh động có thể biến đổi từ 67 ảnh/ giây đến 785 ảnh/ giây. Ở các giá trị trên mức 60 ảnh/ giây hiệu ứng rung thị giác có thể bỏ qua cho hầu hết các mức chiếu sáng.

5.9.2. Hệ số đan xen

Nhằm ngăn chặn hiện tượng rung thị giác ở các tốc độ làm tươi dưới 60 ảnh/ giây, màn hình hiển thị cần phải được đan xen như đã mô tả trong mục 2.3. Độ đan xen tiêu chuẩn cho các hệ thống video là 2:1. Do kỹ thuật đan xen là không cần thiết khi tốc độ làm tươi của màn hình vượt quá 60 ảnh/ giây, nên trong trường hợp đó hệ số đan xen sẽ là 1:1. Nói cách khác, các dòng sẽ được vẽ theo thứ tự thông thường từ dòng 1, 2, 3, 4, ..., cho đến dòng N.

5.9.3. Độ phân giải

Mặc dù được tạo ra từ các vùng có diện tích hữu hạn trong cảnh quan ban đầu (xem mục 5.1 và 5.5), khi được lưu trữ trong bộ nhớ máy tính các pixel vẫn có thể được xem là các điểm toán học không có kích thước vật lý. Tuy nhiên, khi được hiển thị, khoảng cách giữa các điểm này phải được lấp đầy. Quá trình này diễn ra được do sự làm việc của ống CRT với vùng chiếu sáng có kích thước hữu hạn. Phần vùng sáng tạo bởi CRT có phân phối xấp xỉ phân phối Gauss và số lượng của chúng phụ thuộc vào chất lượng của màn hình. Người ta vẫn thường đặc tả một hệ thống hiển thị theo kiểu có độ phân giải 72 vùng sáng/inch (28.3 vùng sáng/cm). Con số này tương ứng với các qui ước chuẩn về in ấn, tuy nhiên trong trường hợp đối tượng hiển thị không phải là máy in thì chúng ta còn có thể sử dụng độ phân giải cao hơn, ví dụ như 30 vùng sáng/cm chẳng hạn.

Chương 6

CÁC PHÉP TOÁN XỬ LÝ ẢNH

Trong chương này chúng ta sẽ tìm hiểu các công cụ toán học giữ vai trò trung tâm trong việc hình thành nên các kỹ thuật xử lý ảnh.

6.1. PHÉP TOÁN CHẬP

Có nhiều ký hiệu biểu diễn phép toán chập giữa hai tín hiệu nhiều chiều, sau đây là cách biểu diễn phổ biến nhất:

$$c = a \otimes b = a * b \quad (33)$$

Chúng ta sẽ sử dụng dạng thứ nhất, $c = a \otimes b$, cho những định nghĩa hình thức tiếp sau trong chương này.

Trong không gian liên tục 2-chiều, phép toán chập được định nghĩa như sau:

$$c(x,y) = a(x,y) \otimes b(x,y) = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} a(\chi, \xi) b * x - \chi, y - \xi d\chi d\xi \quad (34)$$

Còn trong không gian rời rạc 2-chiều, chúng ta có:

$$c[m,n] = a[m,n] \otimes b[m,n] = \sum_{j=-\infty}^{+\infty} \sum_{k=-\infty}^{+\infty} a[j,k] b[m-j, n-k] \quad (35)$$

6.2. CÁC TÍNH CHẤT CỦA PHÉP TOÁN CHẬP

Phép toán chập có một số tính chất toán học quan trọng có nhiều ứng dụng trong xử lý ảnh. Cụ thể là tính giao hoán, tính kết hợp, tính phân phối.

Phép chập có tính *giao hoán*:

$$c = a \otimes b = b \otimes a \quad (36)$$

Phép chập có tính *kết hợp*:

$$c = a \otimes (b \otimes d) = (a \otimes b) \otimes d = a \otimes b \otimes d \quad (37)$$

Phép chập có tính *chất phân phối*:

$$c = a \otimes (b + d) = (a \otimes b) + (a \otimes d) \quad (38)$$

Trong các công thức trên, a, b, c, d là các ảnh liên tục hoặc rời rạc.

6.3. CÁC BIẾN ĐỔI FOURIER

Biến đổi Fourier mang lại một cách biểu diễn khác cho một tín hiệu. Với biến đổi này, tín hiệu sẽ được phân tích thành một tổng có trọng số của những lũy thừa phức. Xuất phát từ công thức Euler:

$$e^{jq} = \cos(q) + j\sin(q) \quad (39)$$

trong đó $j^2 = -1$, chúng ta có thể phát biểu rằng biến đổi Fourier sản sinh ra một biểu diễn tín hiệu 2-chiều dưới dạng một tổng có trọng số các hàm sin và cosin. Nếu cho trước một ảnh a và biến đổi Fourier của nó là A , thì biến đổi Fourier thuận từ miền tọa độ (liên tục hoặc rời rạc) sang miền tần số luôn luôn liên tục.

$$\text{Biến đổi Fourier thuận} - A = F\{a\} \quad (40)$$

Trong khi đó, biến đổi Fourier nghịch lại biến đổi tín hiệu đi từ miền tần số sang miền tọa độ:

$$\text{Biến đổi Fourier nghịch} - a = F^{-1}\{A\} \quad (41)$$

Biến đổi Fourier là phép toán có tính chất duy nhất và có tính khả nghịch sao cho:

$$a = F\{F^{-1}\{A\}\} \quad A = F\{F^{-1}\{a\}\} \quad (42)$$

Sau đây là các công thức cụ thể cho việc chuyển đổi qua lại giữa hai miền tọa độ và tần số theo kiểu biến đổi Fourier.

Trong không gian liên tục 2-chiều chúng ta có:

$$\text{Biến đổi Fourier thuận} - A(u, v) = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} a(x, y) e^{-j(ux+vy)} dx dy \quad (43)$$

$$\text{Biến đổi Fourier nghịch} - a(x, y) = \frac{1}{4\pi^2} \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} A(u, v) e^{j(ux+vy)} du dv \quad (44)$$

Với không gian rời rạc 2-chiều, các công thức trên trở thành:

$$\text{Biến đổi Fourier thuận} - A(\Omega, \Psi) = \sum_{m=-\infty}^{+\infty} \sum_{n=-\infty}^{+\infty} a[m, n] e^{-j(\Omega m + \Psi n)} \quad (45)$$

$$\text{Biến đổi Fourier nghịch} - a[m, n] = \frac{1}{4\pi^2} \int_{-\pi}^{+\pi} \int_{-\pi}^{+\pi} A(\Omega, \Psi) e^{j(\Omega m + \Psi n)} d\Omega d\Psi \quad (46)$$

6.4. CÁC TÍNH CHẤT CỦA BIẾN ĐỔI FOURIER

Có rất nhiều tính chất có liên quan đến phép biến đổi Fourier thuận và nghịch. Sau đây là một vài tính chất có ứng dụng nhiều nhất trong xử lý ảnh.

Một cách tổng quát, biến đổi Fourier là một hàm phức với các biến tần số thực. Do vậy biến đổi này có thể viết lại dưới dạng các thành phần độ lớn và pha của nó như sau:

$$A(u,v) = |A(u,v)| e^{j\varphi(u,v)} \quad A(\Omega,\Psi) = |A(\Omega,\Psi)| e^{j\varphi(\Omega,\Psi)} \quad (47)$$

Một tín hiệu 2-chiều cũng có thể là một tín hiệu phức, do vậy có thể được biểu diễn dưới dạng độ lớn và pha của nó theo công thức sau:

$$a(x,y) = |a(x,y)| e^{j\theta(x,y)} \quad a(m,n) = |a(m,n)| e^{j\theta(m,n)} \quad (48)$$

Nếu tín hiệu 2-chiều là tín hiệu thực, thì biến đổi Fourier của nó có một số tính chất đối xứng cho bởi:

$$A(u,v) = A^*(-u,-v) \quad A(\Omega,\Psi) = A^*(-\Omega,-\Psi) \quad (49)$$

Ký hiệu (*) trong công thức trên thể hiện phép liên hợp phức. Với những tín hiệu thực, phương trình (49) trên trực tiếp dẫn đến:

$$\begin{aligned} |A(u,v)| &= |A(-u,-v)| & \varphi(u,v) &= -\varphi(-u,-v) \\ |A(\Omega,\Psi)| &= |A(-\Omega,-\Psi)| & \varphi(\Omega,\Psi) &= -\varphi(-\Omega,-\Psi) \end{aligned} \quad (50)$$

Nếu một tín hiệu 2-chiều là thực và chẵn, thì biến đổi Fourier của nó cũng có tính thực và chẵn:

$$A(u,v) = A(-u, -v) \quad A(\Omega,\Psi) = A(-\Omega,-\Psi) \quad (51)$$

Biến đổi Fourier thuận và nghịch là các phép toán tuyến tính:

$$F\{w_1 a + w_2 b\} = F\{w_1 a\} + F\{w_2 b\} = w_1 A + w_2 B$$

$$F^{-1}\{w_1 A + w_2 B\} = F^{-1}\{w_1 A\} + F^{-1}\{w_2 B\} = w_1 a + w_2 b \quad (52)$$

trong đó a và b là các tín hiệu 2-chiều (các ảnh số), còn w_1 và w_2 là các hằng số phức tùy ý.

Biến đổi Fourier trong không gian rời rạc, $A(\Omega, \Psi)$, tuần hoàn theo cả hai biến Ω và Ψ với các chu kỳ là 2π :

$$A(\Omega + 2\pi j, \Psi + 2\pi k) = A(\Omega, \Psi) \quad j, k \in \mathbb{Z} \quad (53)$$

Năng lượng E trong một tín hiệu có thể được đo trong cả hai miền tọa độ và tần số. Với một tín hiệu có năng lượng hữu hạn, chúng ta có kết quả sau đây:

Định lý Parseval trong không gian liên tục 2-chiều

$$E = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} |a(x, y)|^2 dx dy = \frac{1}{4\pi} \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} |A(u, v)|^2 du dv \quad (54)$$

Định lý Parseval trong không gian rời rạc 2-chiều

$$E = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} |a(x, y)|^2 dx dy = \frac{1}{4\pi} \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} |A(u, v)|^2 du dv \quad (55)$$

Chúng ta không nên nhầm lẫn dạng “năng lượng tín hiệu” này với năng lượng vật lý của hiện tượng làm phát sinh ra tín hiệu đó. Lấy ví dụ, nếu giá trị $a(m, n)$ biểu diễn cho lượng photon, thì năng lượng vật lý đó sẽ tỷ lệ với độ lớn a , chứ không phải với bình phương của nó. Đây là tình huống thường gặp trong kỹ thuật lấy ảnh video.

Cho trước ba tín hiệu nhiều chiều a, b, c cùng với các biến đổi Fourier A, B, C tương ứng với chúng. Khi đó:

$$\begin{aligned} c = a \otimes b & \quad \xleftarrow{F} C = A \bullet B \\ c = a \bullet b & \quad \xleftarrow{F} C = \frac{1}{4\pi^2} A \otimes B \end{aligned} \quad (56)$$

Nói cách khác, phép toán chập trong miền tọa độ tương đương với phép toán nhân trong miền tần số và ngược lại.

Đây là một kết quả trung tâm không chỉ mang lại cho chúng ta phương pháp luận trong việc cài đặt phép toán chập, mà còn thể hiện rõ cách thức hai tín hiệu tương tác với nhau qua phép chập để cho ra một tín hiệu thứ ba. Chúng ta sẽ sử dụng kết quả này rất nhiều lần nữa trong các chương tiếp sau của giáo trình này.

Nếu một tín hiệu 2-chiều $a(x,y)$ được co giãn theo tọa độ thì:

$$\text{If} \quad a(x, y) \rightarrow a(M_x \bullet x, M_y \bullet y)$$

$$\text{Then} \quad A(u, v) \rightarrow \frac{A\left(\frac{u}{M_x}, \frac{v}{M_y}\right)}{|M_x \bullet M_y|} \quad (57)$$

Nếu một tín hiệu 2-chiều $a(x,y)$ có phổ Fourier $A(u, v)$ thì:

$$A(u=0, v=0) = \int_{-x}^{+x} \int_{-y}^{+y} a(x,y) dx dy$$

$$a(x=0, y=0) = \frac{1}{4\pi^2} \int_{-u}^{+u} \int_{-v}^{+v} A(u, v) du dv \quad (58)$$

$$\frac{\partial a(x, y)}{\partial x} \overset{F}{\leftrightarrow} juA(u, v) \quad \frac{\partial a(x, y)}{\partial y} \overset{F}{\leftrightarrow} jvA(u, v)$$

$$\frac{\partial^2 a(x, y)}{\partial x^2} \overset{F}{\leftrightarrow} -u^2 A(u, v) \quad \frac{\partial^2 a(x, y)}{\partial y^2} \overset{F}{\leftrightarrow} -v^2 A(u, v) \quad (59)$$

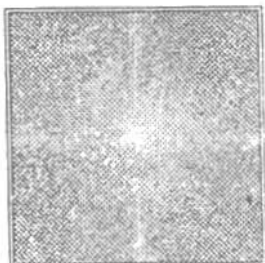
6.4.1 Tầm quan trọng của độ lớn và pha

Phương trình (47) cho thấy rằng biến đổi Fourier của một ảnh số có thể là một hàm phức. Điều này được minh họa dưới đây trong các hình 16 (a) – (c). Hình 16(a) thể hiện ảnh

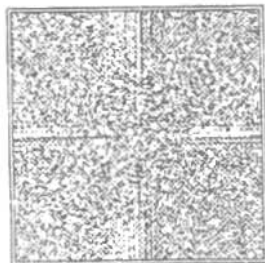
ban đầu $a(m,n)$ hình 16(b) biểu thị giá trị độ lớn dưới dạng tỷ lệ $\log |A(\Omega, \Psi)|$ và hình 16(c) cho biết pha $\varphi(\Omega, \Psi)$.



Hình 16(a)
Ảnh ban đầu

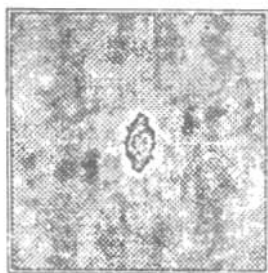


Hình 16(b)
 $\log |A(\Omega, \Psi)|$

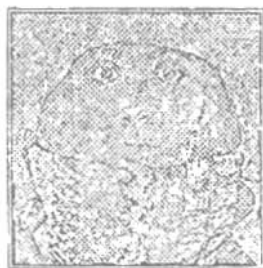


Hình 16(c)
 $\varphi(\Omega, \Psi)$

Cả hai hàm độ lớn và pha đều là các yếu tố cần thiết cho việc tái xây dựng lại hoàn chỉnh một ảnh số từ kết quả biến đổi Fourier của nó. Hình 17(a) cho thấy kết quả thu được khi ảnh gốc ban đầu trong hình 16(a) chỉ được phục hồi dựa vào thông tin độ lớn. Một cách tương tự, hình 17(b) cũng cho thấy kết quả thu được khi ảnh gốc ban đầu trong hình 16(a) chỉ được phục hồi dựa vào thông tin về pha.



Hình 17(a)
 $\varphi(\Omega, \Psi) = 0$



Hình 17(b)
 $|A(\Omega, \Psi)| = \text{constan}$

Như chúng ta có thể quan sát thấy, chỉ riêng thông tin về độ lớn hoặc pha sẽ không đủ hiệu lực để phục hồi lại ảnh

ban đầu. Ảnh chỉ dựa trên độ lớn (hình 17(a)) hoàn toàn không thể nhận ra được và có rất nhiều lỗi nghiêm trọng về thang mức xám động. Ảnh chỉ dựa trên pha (hình 17(b)) về cơ bản vẫn có thể nhận ra được, tuy nhiên bị giảm chất lượng thị giác rất nhiều.

6.4.2. Các tín hiệu đối xứng vòng

Một tín hiệu 2-chiều bất kỳ $a(x,y)$ luôn có thể viết lại dưới dạng tọa độ cực $a(r, \theta)$. Tín hiệu này có tính đối xứng vòng khi quan hệ sau đây xảy ra:

$$a(x,y) = a(r, \theta) = a(r) \quad (60)$$

trong đó $r^2 = x^2 + y^2$ và $tg(\theta) = x/y$. Do một số hệ thống vật lý như các thấu kính lại có tính đối xứng vòng nên chúng ta sẽ đi tìm một biểu diễn Fourier phù hợp cho những tín hiệu như thế.

Một biến đổi Fourier $A(u,v)$ hoàn toàn có thể được biểu diễn trong hệ tọa độ cực dưới dạng $A(\omega, \xi)$. Khi đó nếu tín hiệu có tính đối xứng vòng thì $A(u,v)$ sẽ được biểu diễn dưới dạng một phép biến đổi Hankel cho bởi công thức sau:

$$A(u,v) = F\{a(x,y)\} = 2\pi \int_0^{\infty} a(r) J_0(\omega, r) r dr = A(\omega_r) \quad (61)$$

với: $\omega_r^2 = u^2 + v^2$

$tg\xi = v/u$, $J_0(\cdot)$ là hàm Bessel loại một có bậc 0.

Biến đổi Hankel ngược được xác định bởi:

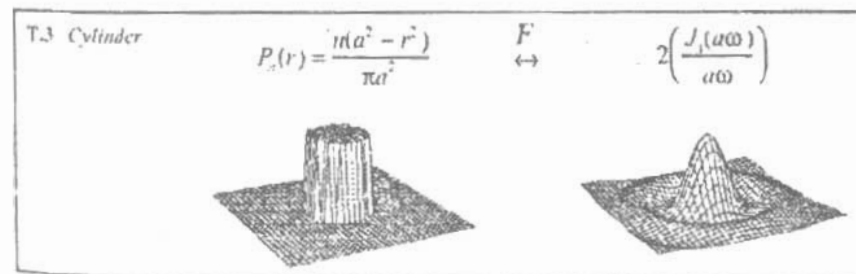
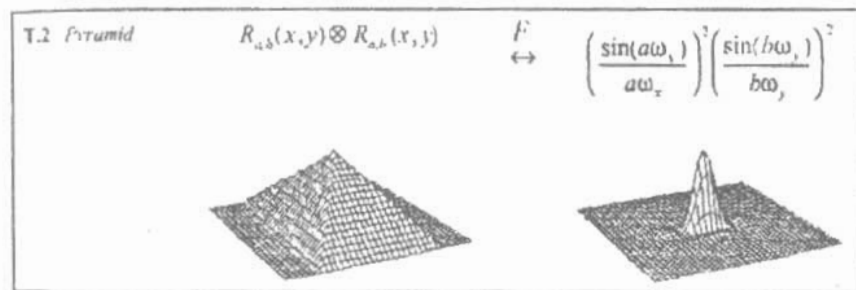
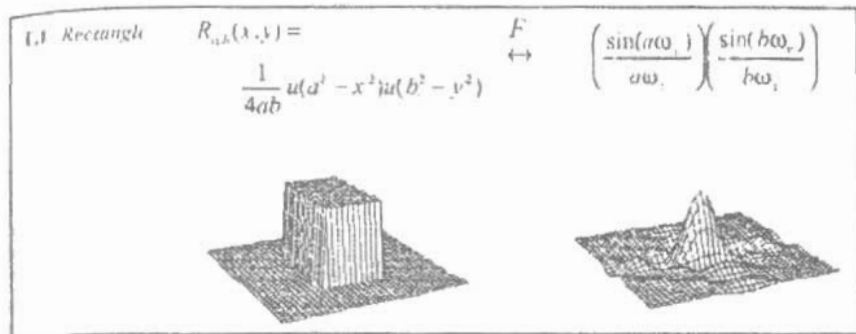
$$a(r) = \frac{1}{2\pi} \int_0^{\infty} A(\omega_r) J_0(\omega, r) \omega_r d\omega_r \quad (62)$$

Biến đổi Fourier của một tín hiệu đối xứng vòng 2-chiều do vậy là một hàm chỉ phụ thuộc vào tần số bán kính ω_r . Sự phụ thuộc vào tần số góc ξ hoàn toàn bị triệt tiêu qua phép biến đổi này. Ngoài ra, nếu $a(x,y) = a(r)$ là một hàm thực thì nó sẽ là hàm chẵn do tính đối xứng vòng. Theo phương trình (51), $A(\omega, r)$ sẽ là hàm thực chẵn.

6.4.3. Ví dụ về các tín hiệu 2-chiều phổ biến cùng với các biến đổi tương ứng của chúng

Bảng 11 cho chúng ta một số tín hiệu cơ bản hữu ích trong xử lý ảnh cùng với các biến đổi Fourier 2-chiều của chúng. Khi sử dụng bảng 11 trong phần còn lại của chương này, chúng ta sẽ xem một miền tọa độ như một *hàm trải điểm* (PSF) hoặc một đáp ứng xung 2-chiều. Còn biến đổi Fourier của nó chính là *hàm chuyển quang học* (OTF), hay đơn giản hơn nữa là *hàm chuyển*. Hai tín hiệu chuẩn được sử dụng trong bảng này là $u(*)$, hàm bước đơn vị, và $J_1(*)$, hàm Bessel loại một. Các tín hiệu đối xứng vòng được biểu diễn dưới dạng các hàm theo r như trong phương trình (60).

Bảng 11: Các ảnh 2-chiều và biến đổi Fourier tương ứng

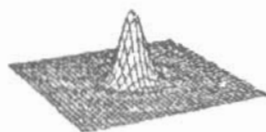


T.4 Cone

$$P_c(r) \otimes P_c(r)$$

F
 \leftrightarrow

$$4 \left(\frac{J_1(a\omega)}{a\omega} \right)^2$$

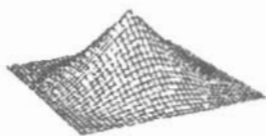


T.5 Airy PSF

$$PSF(r) = \frac{1}{\pi} \left(\frac{J_1(\omega r/2)}{r} \right)^2$$

F
 \leftrightarrow

$$\frac{2}{\pi} \left\{ \cos^{-1} \left(\frac{\omega}{\omega_c} \right) - \left(\frac{\omega}{\omega_c} \right) \sqrt{1 - \left(\frac{\omega}{\omega_c} \right)^2} \right\} u(\omega_c^2 - \omega^2)$$



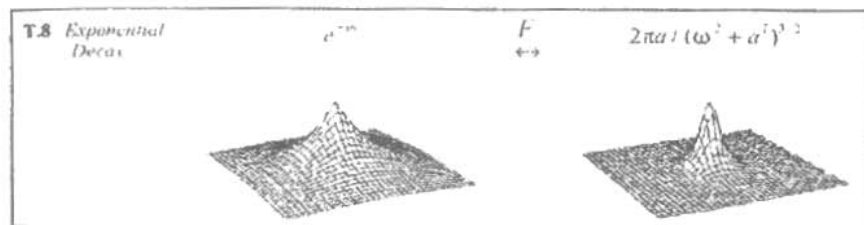
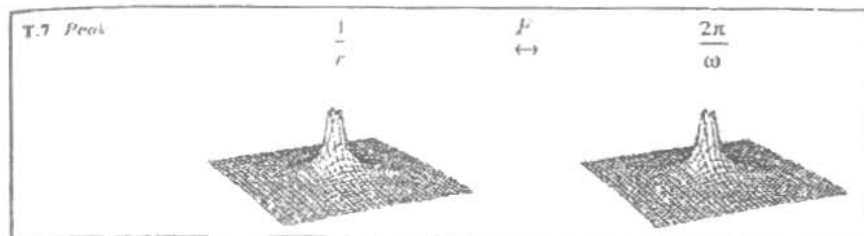
T.6 Gaussian

$$g_{2D}(r, \sigma) = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{r^2}{2\sigma^2}\right)$$

F
 \leftrightarrow

$$G_{2D}(f, \sigma) = \exp(-\omega^2 \sigma^2 / 2)$$





6.5. CÁC MÔ TẢ THỐNG KÊ

Trong xử lý ảnh, việc sử dụng các mô tả thống kê đơn giản cho ảnh và những ảnh con của nó là hết sức phổ biến. Những mô tả này có mối liên hệ mật thiết với khái niệm phân phối xác suất, đặc biệt là phân phối độ lớn của tín hiệu. Với một vùng ảnh cho trước, và đôi khi là toàn bộ ảnh, chúng ta có thể định nghĩa hàm phân phối xác suất và hàm trừ mật xác suất về độ sáng cho vùng đó. Trong những phần tiếp theo đây, chúng ta giả sử rằng đang làm việc với một ảnh số hóa $a[m,n]$.

6.5.1. Hàm phân phối xác suất về độ sáng

Hàm phân phối xác suất $P(a)$ thể hiện xác suất chọn được một giá trị độ sáng từ một vùng ảnh cho trước, sao cho giá trị này không vượt quá một giá trị sáng cho trước a . Khi a biến thiên từ $-\infty$ đến $+\infty$, $P(a)$ sẽ nhận các giá trị từ 0 đến 1. $P(a)$ là hàm đơn điệu, không giảm theo a và do vậy $dP/da \geq 0$.

6.5.2 Hàm trừ mật xác suất về độ sáng

Xác suất để một độ sáng trong một vùng rơi vào khoảng giữa hai giá trị a và $a + \Delta a$, trong trường hợp biết trước hàm phân phối xác suất $P(a)$, có thể được biểu diễn dưới dạng $p(a)\Delta a$, trong đó $p(a)$ là hàm trừ mật xác suất được định nghĩa theo công thức sau:

$$p(a)\Delta a = \left(\frac{dP(a)}{da} \right) \Delta a \quad (63)$$

Do tính đơn điệu không giảm của $P(a)$, chúng ta có:

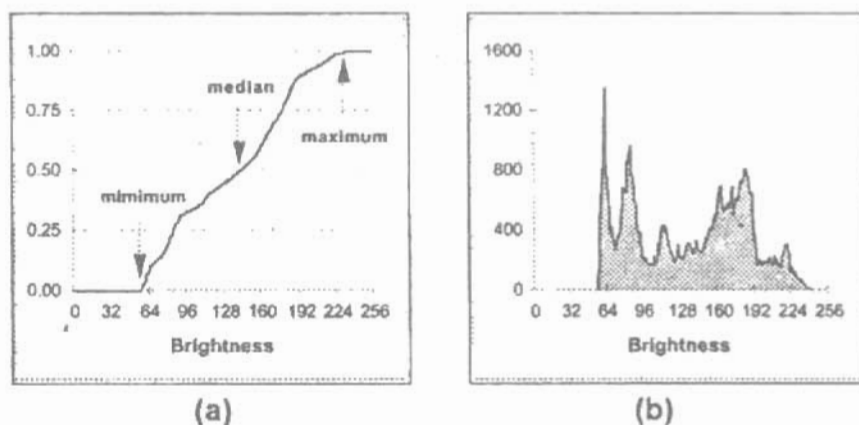
$$p(a) \geq 0; \quad \int_{-\infty}^{+\infty} p(a) da = 1 \quad (64)$$

Với ảnh có giá trị độ sáng đã được lượng hóa (giá trị nguyên), đại lượng Δa trong các công thức nói trên biểu thị cho bề rộng của một khoảng độ sáng, và thường được giả định là không đổi cho toàn vùng ảnh khảo sát. Hàm trừ mật xác suất về độ sáng thường được ước lượng bằng cách đếm số lần một giá trị độ sáng xuất hiện trong vùng ảnh để sinh ra được một lược đồ $h[a]$. Lược đồ này sau đó có thể được chuẩn hóa sao cho tổng diện tích của phần bên dưới lược đồ là 1 (phương trình (64)). Nói cách khác, giá trị $p[a]$ của một vùng chính là số pixel A đã được chuẩn hóa của vùng đó, trong một vùng có độ sáng đã được lượng hóa a :

$$p(a) = \frac{1}{A} h[a]; \quad A = \sum_a h[a] \quad (65)$$

Hàm phân phối xác suất của ảnh có trong hình 16(a) được thể hiện trên hình 18(a). Phần lược đồ sáng chưa chuẩn hóa trong hình 16(a) tỷ lệ với hàm trừ mật xác suất đã được

ước lượng có thể tìm thấy trong hình 18(b). Chiều cao của mỗi cột trong lược đồ này bằng với số pixel có giá trị độ sáng ngay tại vị trí cột đó.



Hình 18: (a) Hàm phân phối xác suất về độ sáng (hình 16(a)) với các giá trị nhỏ nhất, trung bình, và lớn nhất. (b) Lược đồ độ sáng của hình 16(a).

Cả hai đại lượng hàm phân phối và lược đồ thu được từ một vùng ảnh đều có thể xem là một mô tả thống kê về vùng đó. Chúng ta cần phải nhấn mạnh ở đây là cả $P[a]$ và $p[a]$ chỉ nên được xem như những ước lượng cho một phân phối xác suất thực sự ứng với một vùng ảnh cụ thể. Điều này có nghĩa là chúng ta quan sát một ảnh và xem một vùng cụ thể trong ảnh là một hiện thực hóa của nhiều quá trình ngẫu nhiên khác nhau tham gia vào việc hình thành nên ảnh và vùng đó. Với cùng ngữ cảnh như vậy, các đại lượng thống kê được định nghĩa sau đây cũng chỉ nên xem là các ước lượng cho những tham số trong một mô hình thống kê thực sự.

6.5.3 Đại lượng trung bình

Giá trị độ sáng trung bình của một vùng ảnh được xác định bởi *trung bình mẫu* của độ sáng của pixel thuộc vùng đó. Nếu ký hiệu m là độ sáng trung bình của A pixel trong một vùng \mathcal{A} thì

$$m_a = \frac{1}{\Lambda} \sum_{(m,n) \in \mathcal{A}} h[a] \quad (66)$$

Chúng ta còn có thể sử dụng một công thức khác cho độ sáng trung bình dựa vào lược đồ sáng chưa chuẩn hóa $h[a] = A \cdot p(a)$, với các giá trị sáng rời rạc a . Khi đó

$$m_a = \frac{1}{\Lambda} \sum_a a \cdot h[a] \quad (67)$$

Giá trị độ sáng trung bình m_a nói trên chính là một ước lượng của độ sáng kỳ vọng u_a trong phân phối xác suất về độ sáng.

6.5.4. Độ lệch chuẩn

Ước lượng không thiên vị của độ lệch chuẩn cho các giá trị sáng trong một vùng \mathcal{A} gồm Λ pixel, ký hiệu bởi s_a , được gọi là *độ lệch chuẩn mẫu*. Độ lệch này được xác định theo công thức sau:

$$\begin{aligned} s_a &= \sqrt{\frac{1}{\Lambda - 1} \sum_{(m,n) \in \mathcal{A}} (a[m,n] - m_a)^2} \\ &= \sqrt{\frac{\sum_{(m,n) \in \mathcal{A}} a^2[m,n] - \Lambda m_a^2}{\Lambda - 1}} \end{aligned} \quad (68)$$

Theo công thức lược đồ chúng ta có:

$$s_a = \sqrt{\frac{\left(\sum_a a^2 \cdot h[a] \right) - \Lambda \cdot m_a^2}{\Lambda - 1}} \quad (69)$$

Giá trị độ lệch chuẩn mẫu s_a nói trên chính là một ước lượng của độ lệch chuẩn σ_a trong phân phối xác suất về độ sáng.

6.5.5 Hệ số biến đổi

Hệ số biến đổi không có thứ nguyên CV (Coefficient of Variation) được xác định bởi:

$$CV = \frac{s_a}{m_a} \times 100\% \quad (70)$$

6.5.6. Đại lượng phần trăm

Giá trị phần trăm $p\%$ của một phân phối độ sáng chưa được lượng hóa được xác định theo mỗi giá trị độ sáng a sao cho:

$$P(a) = p\%$$

hay một cách tương đương:

$$\int_{-\infty}^a p(\alpha) d\alpha = p\% \quad (71)$$

Có ba trường hợp đặc biệt sau đây của đại lượng phần trăm thường được sử dụng rất nhiều trong xử lý ảnh:

- Giá trị 0% biểu thị giá trị nhỏ nhất trong một vùng.
- Giá trị 50% biểu thị giá trị trung bình trong một vùng.
- Giá trị 100% biểu thị giá trị lớn nhất trong một vùng.

Cả ba giá trị này đều có thể xác định được trong hình 18(a).

6.5.7. Tỷ số tín hiệu – nhiễu

Tỷ số tín hiệu – nhiễu SNR (Signal – To – Noise Ratio) có nhiều cách định nghĩa khác nhau. Nhiễu ở đây được đặc trưng bằng độ lệch chuẩn s_n của nó, trong khi đó đặc trưng của tín hiệu sẽ được thể hiện dưới nhiều hình thức khác. Nếu tín hiệu a nằm giữa hai biên $a_{min} \leq a \leq a_{max}$ thì đại lượng SNR sẽ được xác định bởi:

$$\text{Tín hiệu bị chặn} - SNR = 20 \log_{20} \left(\frac{a_{max} - a_{min}}{s_n} \right) \text{dB} \quad (72)$$

Trong trường hợp tín hiệu không bị chặn nhưng có phân phối thống kê thì có hai định nghĩa phổ biến sau đây cho SNR .

Tín hiệu ngẫu nhiên – Tín hiệu và nhiễu phụ thuộc nhau

$$SNR = 20 \log_{20} \left(\frac{m_a}{s_n} \right) \text{dB} \quad (73)$$

$$\text{Tín hiệu và nhiễu độc lập nhau} \quad SNR = 20 \log_{20} \left(\frac{s_a}{s_n} \right) \text{dB} \quad (74)$$

trong đó m_a và s_a được định nghĩa như trên.

Bảng 12 dưới đây liệt kê các giá trị thống kê ứng với ảnh và vùng đã được chỉ ra trong hình 19.

Việc tính giá trị SNR cho toàn bộ ảnh dựa trên phương trình (72) không có sẵn trong bảng nói trên. Những thay đổi lớn trong độ sáng ảnh dẫn đến giá trị lớn của s ($s = 49.5$) nhìn chung không phải do nhiễu gây ra, mà do sự thay đổi của thông tin cục bộ. Với khái niệm vùng, chúng ta luôn có thể ước lượng được giá trị SNR . Chúng ta có thể sử dụng giá trị $s_{\eta} = 4.0$ và thang độ xám động $a_{max} - a_{min}$ (241 – 56) cho

ảnh để tính được giá trị SNR toàn cục (33.3dB). Ở đây có hai giả định là:

(1) Tín hiệu gần như không đổi và sự thay đổi trong vùng là do nhiễu gây ra.

(2) Nhiễu giống nhau trên toàn ảnh và được đặc trưng bởi độ lệch chuẩn $s_n = s_{\sigma}$



Hình 19. Vùng khảo sát ở đây thuộc phạm vi bên trong vòng tròn

Bảng 12. Các giá trị thống kê được lấy từ hình 19

Các giá trị thống kê	Ảnh	Vùng quan tâm
Trung bình	137.7	219.3
Độ lệch chuẩn	49.5	4.0
Giá trị nhỏ nhất	56	202
Giá trị trung bình	141	220
Giá trị lớn nhất	241	226
SNR (db)	Không tính	33.3

6.6. BIỂU DIỄN BIÊN CỦA ĐỐI TƯỢNG

Khi làm việc với một vùng hoặc một đối tượng, chúng ta có rất nhiều cách biểu diễn ngắn gọn giúp ích cho sự thao tác và đo đạc chúng. Sau đây là một số cách biểu diễn thông dụng thông qua việc mô tả đường biên của vùng hay đối tượng.

6.6.1. Mã xích

Cách biểu diễn này dựa trên công trình của Freeman trong [11]. Với cách biểu diễn này, chúng ta lần theo biên của một vùng hay đối tượng theo chiều kim đồng hồ và lưu lại hướng đi mỗi khi chúng ta chuyển từ một pixel biên này sang một pixel biên khác. Với cách cài đặt mã xích chuẩn, một pixel biên sẽ được xem là một pixel thuộc đối tượng khi một trong các pixel thuộc lân cận 4 của nó lại là một pixel nền (xem hình 4(a) và 20 (c)).

Các mã được sử dụng trong cách biểu diễn này liên kết với 8 hướng di chuyển có thể có khi đi từ một pixel sang một pixel khác kề nó và được định nghĩa như sau:

$$\begin{array}{ccc} 3 & 2 & 1 \\ 4 & x & 0 \\ 5 & 6 & 7 \end{array} \quad (75)$$

Trong định nghĩa trên, x thể hiện vị trí hiện thời của điểm biên mà chúng ta đang di chuyển theo.

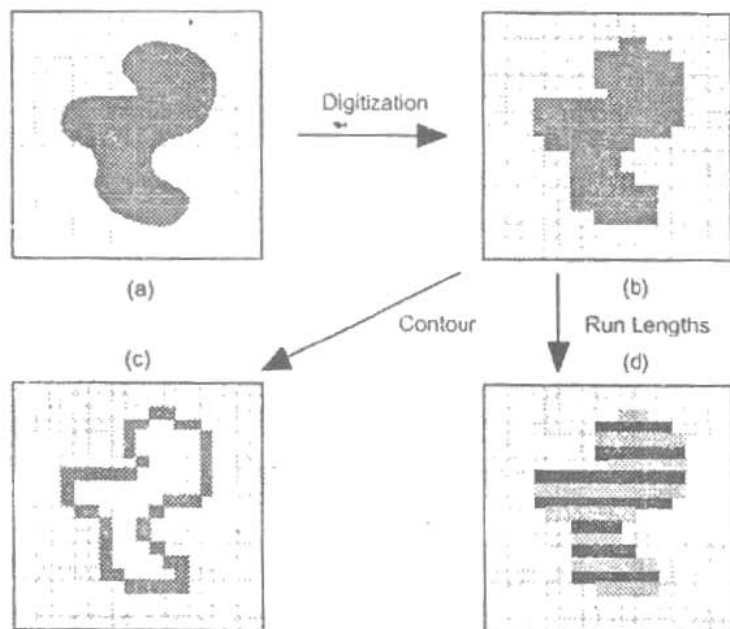
6.6.2. Các tính chất của mã xích

Các mã chẵn $[0, 2, 4, 6]$ tương ứng với các hướng di chuyển ngang và dọc trong ảnh, còn các mã lẻ $[1, 3, 5, 7]$ tương ứng với những hướng di chuyển chéo.

Mỗi mã có thể được xem như là một hướng góc (bội số của 45°) cần phải rẽ khi di chuyển từ một điểm biên này sang một điểm biên khác.

Các tọa độ tuyệt đối $[m,n]$ của pixel biên đầu tiên cùng với các mã xích của biên sẽ thể hiện một mô tả đầy đủ về biên của một vùng rời rạc.

Khi có sự thay đổi giữa hai mã xích liên tiếp, chúng ta sẽ có sự đổi hướng trên biên của vùng đang xét. Điểm ảnh có tính chất này được gọi là *một góc* của đường biên.



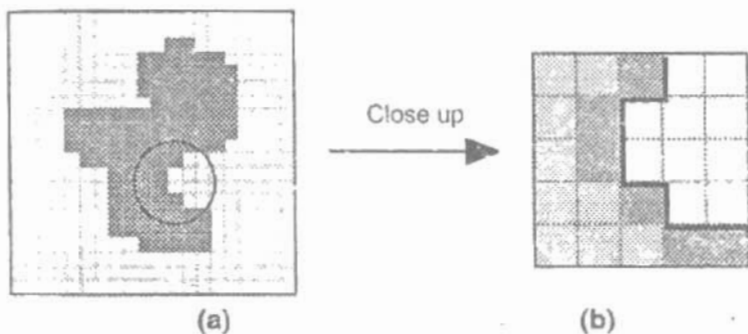
Hình 20. Vùng ảnh (phân tô bóng) được biến đổi từ (a) dạng liên tục sang (b) dạng rời rạc, và có thể được xem như (c) một đường biên, hoặc (d) các độ dài đường chạy được minh họa bằng các dây màu luân phiên.

6.6.3. Mã crack

Chúng ta còn có một lựa chọn khác cho việc biểu diễn biên của một vùng hay đối tượng mà không cần sử dụng đến những pixel biên liên kết với đối tượng cũng như với nền ảnh. Ở đây chúng ta chỉ sử dụng đường ranh giới giữa chúng, như được minh họa trong hình 21 với một phần phóng lớn của hình 20.

Mã “crack” có thể được xem như mã xích với 4 hướng thay vì 8 hướng, theo như định nghĩa sau đây:

$$\begin{array}{ccc} & 1 & \\ 2 & x & 0 \\ & 3 & \end{array} \quad (76)$$



Hình 21. (a) Đối tượng chứa phần ảnh cần khảo sát (b) Các pixel biên sử dụng trong mã xích được tô gạch chéo, còn phần mã “crack” được chỉ ra bằng đường tô đen dày trong hình.

Mã xích cho phần phóng lớn của hình 21 (b) từ trên xuống dưới là $[5, 6, 7, 7, 0]$. Mã crack tương ứng là $[3, 2, 3, 3, 0, 3, 0, 0]$.

6.6.4. Mã chạy

Cách biểu diễn đối tượng phổ biến sau cùng được xây dựng dựa trên cơ sở mã hóa dãy các pixel thuộc về một đối tượng và nằm liên tiếp nhau trên một hàng. Dãy những pixel như vậy được gọi là một hàng chạy và được xác định bởi các vị trí bắt đầu và kết thúc của hàng đó. Hình 20(d) cho chúng ta một số ví dụ về những hàng chạy như vậy. Trong biểu diễn này, chúng ta cần phải có các định nghĩa chính xác về vị trí bắt đầu và kết thúc của mỗi hàng chạy. Những định nghĩa như vậy thông thường phụ thuộc vào từng ứng dụng cụ thể, nên chúng ta sẽ không trình bày chúng ở đây.

6.7. CÁC PHÉP TOÁN TRÊN LƯỢC ĐỒ (HISTOGRAM)

Trong xử lý ảnh, chúng ta có một lớp quan trọng các phép toán trên điểm dựa vào sự thao tác trên lược đồ của ảnh hoặc của một vùng trong ảnh. Những phép toán phổ biến nhất trong số đó sẽ được trình bày tiếp sau đây.

6.7.1. Phép co giãn độ tương phản

Thông thường, một ảnh số được quét vào trong máy tính từ một máy quét sẽ có các giá trị độ sáng không tận dụng được đầy đủ thang độ sáng có trong thực tế. Chúng ta dễ dàng quan sát thấy điều này bằng cách xem xét lược đồ các giá trị sáng như đã được trình bày trong hình 18. Để thay đổi tình thế này, chúng ta có thể tiến hành co giãn lược đồ trên thang độ sáng động của ảnh. Nếu độ sáng trong ảnh có khuynh hướng trải từ 0 đến $2^B - 1$ (xem mục 2.1), khi đó chúng ta ánh xạ giá trị 0% (hay còn gọi là giá trị *cực tiểu* như đã định nghĩa trong mục 6.5.2) sang giá trị 0 và giá trị 100%

(hay giá trị *cực đại*) sang giá trị $2^B - 1$. Phép biến đổi thích hợp được cho dưới đây.

$$b[m,n] = (2^B - 1) \frac{a[m,n] - \text{cực_thiểu}}{\text{cực_đại} - \text{cực_thiểu}} \quad (77)$$

Một phiên bản tổng quát hơn của công thức trên là:

$$b[m,n] = \begin{cases} 0 & a[m,n] \leq p_{\text{low}}\% \\ (2^B - 1) \cdot \frac{a[m,n] - p_{\text{low}}\%}{p_{\text{high}}\% - p_{\text{low}}\%} & p_{\text{low}}\% < a[m,n] < p_{\text{high}}\% \\ (2^B - 1) & a[m,n] \geq p_{\text{high}}\% \end{cases} \quad (78)$$

Trong phiên bản thứ hai này chúng ta có thể chọn các giá trị 1% và 99% cho $p_{\text{low}}\%$ và $p_{\text{high}}\%$, thay vì phải chọn các giá trị 0% và 100% như trong phương trình (77). Chúng ta còn có thể áp dụng phép toán này cho một vùng ảnh, bằng cách xét lược đồ của vùng đó nhằm tìm ra những giới hạn thích hợp cho thuật toán. Cần chú ý rằng trong hai phương trình (77) và (78) chúng ta có thể triệt tiêu số hạng $2^B - 1$ và chuẩn hóa thang độ sáng về khoảng $0 \leq b[m, n] \leq 1$. Điều này có nghĩa là giá trị độ sáng của các điểm ảnh sẽ được biểu diễn dưới dạng số thực thay vì số nguyên. Khả năng của CPU và dung lượng RAM của các máy tính hiện tại hoàn toàn có thể đáp ứng được cách biểu diễn này.

6.7.2. Cân bằng lược đồ

Để so sánh hai hay nhiều ảnh khác nhau, trước hết các lược đồ của những ảnh này thường được chuyển đổi về một

dạng lược đồ chuẩn. Việc làm này, với tên gọi là *chuẩn hóa lược đồ*, đặc biệt hữu ích khi các ảnh được thu nhận từ nhiều tình huống khác nhau. Kỹ thuật chuẩn hóa lược đồ ảnh phổ biến nhất là *cân bằng lược đồ*, trong đó lược đồ ban đầu được biến đổi nhờ một hàm $b = f(a)$ sang một lược đồ mới có các giá trị độ sáng không đổi. Về mặt lý thuyết, lược đồ mới này tương ứng với một phân phối độ sáng, trong đó các giá trị sáng có xác suất như nhau. Tuy nhiên với một ảnh bất kỳ, chúng ta chỉ có thể đạt được sự xấp xỉ kết quả này chứ không thể đạt được độ chính xác hoàn toàn như lý thuyết.

Với một hàm “thích hợp” $f(\bullet)$ thì mối quan hệ giữa hàm trừ mật xác suất đầu vào, hàm trừ mật xác suất đầu ra, và hàm $f(\bullet)$ sẽ được cho bởi:

$$p_b(b)db = p_a(a)da \Rightarrow df = \frac{p_a(a)da}{p_b(b)} \quad (79)$$

Từ phương trình (79), chúng ta thấy rằng từ “thích hợp” trên đây hàm ý $f(\bullet)$ khả vi và do vậy $df/da \geq 0$. Để cân bằng lược đồ, chúng ta mong muốn rằng $p_b(b) = \text{hằng số}$ và điều này có nghĩa là:

$$f(a) = (2^B - 1) \cdot P(a) \quad (80)$$

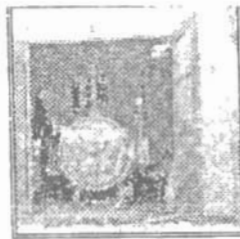
với $P(a)$ là hàm phân phối xác suất đã định nghĩa trong mục 6.5.1 và minh họa trong hình 18(a). Nói cách khác, hàm phân phối xác suất đã *lượng hóa* được chuẩn hóa từ 0 đến $2^B - 1$ là một bảng tra cần thiết cho việc cân bằng lược đồ. Hình 22(a)–(c) minh họa tác động của việc cơ giãn lược đồ và cân bằng lược đồ trên một ảnh chuẩn. Thủ tục cân bằng lược đồ này cũng có thể áp dụng trên cơ sở từng vùng ảnh một.



Hình 22(a)
Ảnh ban đầu



Hình 22(b)
Ảnh đã có giãn độ
tương phản



Hình 22(c)
Ảnh đã cân bằng
lược đồ

6.7.3. Các phép toán dựa trên lược đồ khác

Những lược đồ được lấy từ các vùng cục bộ cũng có thể được dùng để tạo ra các bộ lọc cục bộ áp dụng cho những vùng đó. Các ví dụ cụ thể bao gồm *lọc cục tiểu*, *lọc trung vị*, và *lọc cục đại* [23]. Các khái niệm cục tiểu, cục đại, và trung vị đã được giới thiệu trong hình 18. Các bộ lọc dựa trên những khái niệm này sẽ được trình bày trong mục 6.10.2 và 6.12.10.

6.8. CÁC PHÉP TOÁN TOÁN HỌC

Trong mục này chúng ta phân biệt hai kiểu số học là số học bình thường và số học nhị phân. Trong trường hợp nhị phân, chúng ta chỉ có hai giá trị sáng phân biệt là “0” và “1”. Trong trường hợp thông thường, chúng ta bắt đầu với 2^B giá trị (mức) sáng nhưng việc xử lý ảnh dễ dàng sinh ra thêm nhiều mức nữa. Vì lý do như vậy, nhiều hệ thống phần mềm cung cấp các biểu diễn 16 hay 32-bit cho giá trị độ sáng của điểm ảnh nhằm tránh những vấn đề sẽ gặp phải với việc tràn số học.

6.8.1. Các thao tác nhị phân

Các thao tác nhị phân tạo thành một cơ sở vững chắc cho những công cụ hết sức mạnh mẽ được mô tả ở đây và sẽ

được mở rộng trong mục 6.12, khi chúng ta thảo luận về lý thuyết morphology toán học cùng với những ứng dụng của nó trong xử lý ảnh. Những thao tác được nêu ra dưới đây là các thao tác trên điểm, do vậy chúng có rất nhiều cách cài đặt hiệu quả, bao gồm cả cách cài đặt bằng các bảng tra đơn giản. Ký hiệu chuẩn cho tập các thao tác nhị phân cơ bản là:

$$NOT \quad c = \bar{a}$$

$$OR \quad c = a + b$$

$$AND \quad c = a \cdot b$$

$$XOR \quad c = a \oplus b = a \cdot \bar{b} + \bar{a} \cdot b$$

$$SUB \quad c = a \setminus b = a - b = a \cdot \bar{b}$$

Ở đây mỗi thao tác sẽ được áp dụng cho từng điểm ảnh một. Lấy ví dụ, $c[m,n] = a[m,n] \cdot \bar{b}[m,n]$, $\forall m, n$. Định nghĩa của mỗi thao tác được cho sau đây.

NOT a	
0	1
1	0

OR a	B	
	0	1
0	0	1
1	1	1

AND a	b	
	0	1
0	0	0
1	0	1

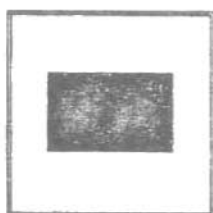
XOR a	b	
	0	1
0	0	1
1	1	0

(82)

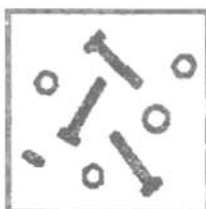
SUB a	B	
	0	1
0	0	0
1	1	0

(83)

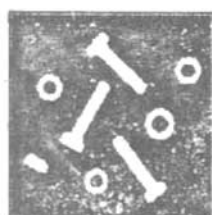
Hình 23 minh họa những thao tác nói trên với giá trị nhị phân “1” có màu đen, còn giá trị nhị phân “0” có màu trắng.



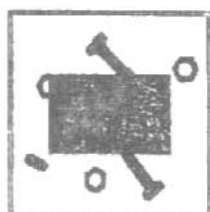
a) Ảnh a



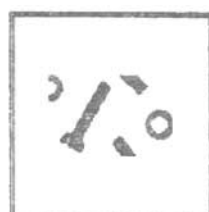
b) Ảnh b



c) $\text{NOT}(b) = \bar{b}$



d) $\text{OR}(a,b) = a + b$



e) $\text{AND}(a,b) = a * b$



f) $\text{XOR}(a,b) = a \oplus b$



g) $\text{SUB}(a,b) = a \setminus b$

Hình 23. Các ví dụ về những thao tác điểm nhị phân

Phép toán $\text{SUB}(*)$ sẽ rất hữu ích khi ảnh a thể hiện vùng quan tâm mà chúng ta muốn phân tích một cách có hệ thống, còn ảnh b thể hiện các đối tượng đã được phân tích và có thể loại đi khỏi vùng đó.

6.8.2. Các thao tác số học

Các thao tác trên điểm ảnh có trị xám được xây dựng dựa trên các phép toán số học thông thường cho trong bảng sau:

Thao tác	Định nghĩa	Kiểu dữ liệu mong muốn
ADD	$c = a + b$	Số nguyên
SUB	$c = a - b$	Số nguyên
MUL	$c = a * b$	Số nguyên hoặc số thực
DIV	$c = a / b$	Số thực
LOG	$c = \log(a)$	Số thực
EXP	$c = \exp(a)$	Số thực
SQRT	$c = \text{sqrt}(a)$	Số thực
TRIG.	$c = \sin/\cos/\tan(a)$	Số thực
INVERT	$c = (2^8 - 1) - a$	Số nguyên

6.9. CÁC PHÉP TOÁN DỰA TRÊN PHÉP CHẬP

Phép chập, về mặt toán học, là một phép toán cục bộ đã được định nghĩa trong mục 6.1. Nó là phép toán trung tâm của lý thuyết xử lý ảnh hiện đại với ý tưởng cơ bản có thể trình bày ngắn gọn như sau: chúng ta có một cửa sổ với kích thước hữu hạn và một hình dạng nhất định (còn gọi là giá đỡ). Cửa sổ này được sử dụng để quét ngang qua ảnh. Khi đó, giá trị điểm ảnh đầu ra sẽ là tổng có trọng số của những giá trị điểm ảnh đầu vào nằm bên trong cửa sổ. Ở đây các trọng số là các giá trị của bộ lọc được gán cho từng điểm ảnh của cửa sổ đó. Cửa sổ vừa nêu cùng với những trọng số của nó

được gọi là *nhân chập*. Điều này ngay lập tức dẫn đến biến thể sau đây của phương trình (35). Nếu bộ lọc – có trị 0 ở bên ngoài của sổ chữ nhật – thì bằng cách sử dụng phương trình (68), phép chập có thể được viết lại dưới tổng hữu hạn sau:

$$c[m,n] = a[m,n] \otimes h[m,n] = \sum_j \sum_{k=0}^{K-1} h[j,k] a[m-j,n-k] \quad (84)$$

Ngoài việc cung cấp một cơ chế quan trọng cho sự làm trơn hay làm sắc nét ảnh, phương trình trên còn có nhiều ứng dụng khác. Hơn nữa, trong khi phương trình (84) minh họa cho tính chất cục bộ của thao tác chập, thì các phương trình 74) và (56) cho thấy rằng nó có thể được cài đặt thông qua miền Fourier, vốn đòi hỏi một phép toán toàn cục là biến đổi Fourier. Những khía cạnh vừa nêu của phép chập sẽ được trình bày trong các mục tiếp sau đây.

6.9.1. Kiến thức nền

Trong rất nhiều hệ thống hình thành ảnh, một mô hình thích hợp cho việc biến đổi một tín hiệu vật lý $a(x,y)$ thành một tín hiệu điện tử $c(x,y)$ chính là phép chập tín hiệu đầu vào với đáp ứng xung của hệ thống cảm biến. Một hệ thống như vậy có thể bao gồm cả hai hệ thống con quang học và điện tử. Nếu mỗi hệ con này có thể xem là một hệ thống tuyến tính bất biến dịch chuyển (*Linear Shift Invariant*) thì khi đó mô hình chập là phù hợp. Các định nghĩa có liên quan đến những tính chất của hai hệ thống con như vậy được cho dưới đây:

$$\text{if } a_1 \rightarrow c_1 \text{ và } a_2 \rightarrow c_2$$

Tính tuyến tính –

$$\text{then } w_1 \bullet a_1 + w_2 \bullet a_2 \rightarrow w_1 \bullet c_1 + w_2 \bullet c_2$$

Tính bất biến dịch chuyển –

if $a(x,y) \rightarrow c(x,y)$

$$\text{then } a(x - x_0, y - y_0) \rightarrow c(x - x_0, y - y_0) \quad (86)$$

trong đó w_1 và w_2 là các hằng số phức tùy ý, còn x_0 và y_0 là các tọa độ tương ứng với một phép tịnh tiến bất kỳ nào đó.

Đến đây có hai ghi nhận mà chúng ta cần phải lưu ý. Ghi nhận đầu tiên là, tính tuyến tính của hệ thống cho phép kết luận rằng (bằng cách chọn $w_1 = w_2 = 0$) nếu có “giá trị 0 vào” thì cũng có “giá trị 0 ra”. Độ dịch chuyển đã được mô tả trong phương trình (58) cho thấy rằng tín hiệu camera như vậy không phải là tín hiệu đầu ra của một hệ thống tuyến tính.

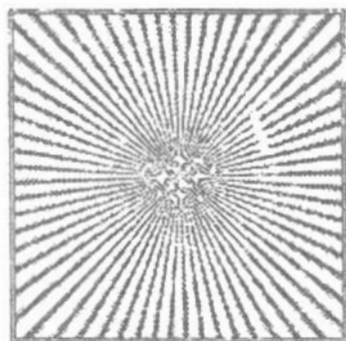
Ghi nhận thứ hai là, các thấu kính quang học với độ phóng đại M khác 1x không có tính bất biến dịch chuyển, do vậy sự tịnh tiến đi 1 đơn vị trong ảnh đầu vào $a(x,y)$ sẽ làm sản sinh ra sự tịnh tiến M đơn vị trong ảnh đầu ra $c(x,y)$. Do tính chất Fourier đã được mô tả trong phương trình (57), tình huống này vẫn có thể quản lý được thông qua lý thuyết các hệ tuyến tính.

Nếu một điểm sáng xung $\delta(x,y)$ được lấy ảnh thông qua một hệ thống LSI thì đáp ứng xung của hệ thống đó sẽ được gọi là *hàm trải điểm (Point Spread Function)*. Khi đó ảnh đầu ra trở thành kết quả của việc chập ảnh đầu vào với PSF. Biến đổi Fourier của PSF được gọi là *hàm chuyển quang học (Optical Transfer Function)*. Với những hệ thống quang học có tính đối xứng vòng, không có quang sai, và có độ nhiễu xạ giới hạn, PSF được cho bởi đĩa Airy trong bảng 11-T.5. OTF của đĩa Airy cũng được thể hiện trong bảng 11-T.5.

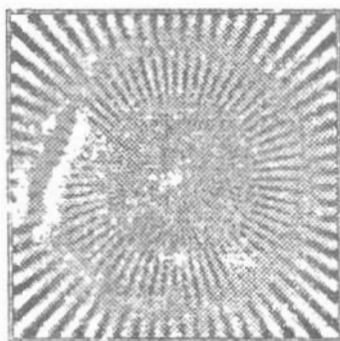
Nếu cửa sổ chập không phải là PSF có nhiễu xạ giới hạn của các thấu kính, mà là kết quả tác động của việc làm phân

kỳ một thấu kính, thì một mô hình thích hợp cho $h(x,y)$ chính là một bao nang thuốc có bán kính a như đã mô tả trong bảng 11 – T.6. Tác động của một mẫu kiểm tra được minh họa trên hình 24.

Tác động của sự phân kỳ không chỉ đơn thuần là làm cho ảnh đầu vào mờ đi hoặc trơn hơn. Các mẫu âm gần như tuần hoàn trong hàm chuyển ở bảng 11–T.3 cho ra một độ dịch chuyển pha 180° , trong đó đen chuyển thành trắng và ngược lại. Trong hình 24(b), chúng ta có thể quan sát thấy sự dịch chuyển pha đó hết sức rõ ràng.



(a) Mẫu kiểm tra



(b) Ảnh phân kỳ

Hình 24. Phép chụp mẫu kiểm tra với một bao nang thuốc có bán kính $a = 4.5$ điểm ảnh.

6.9.2. Phép chụp trong miền tọa độ

Trong khi mô tả các bộ lọc dựa trên phép chụp chúng ta sẽ sử dụng qui ước sau đây. Cho trước một bộ lọc $h[j,k]$ có kích thước $J \times K$, chúng ta xem như tọa độ $[j = 0, k = 0]$ là tâm của ma trận lọc h . Ý tưởng này được minh họa trong hình 25. “Tâm” của ma trận được xác định rõ ràng khi J và K đều lẻ.

Khi cả hai đều là các số chẵn, chúng ta sẽ sử dụng các xấp xỉ $(J/2, K/2)$ cho “tâm” của ma trận.

Khi khảo sát tổng chập theo phương trình (84) một cách kỹ lưỡng hơn, chúng ta sẽ giải thích được nhiều vấn đề phức tạp, như được trình bày tiếp sau đây.

$$h = \begin{bmatrix} h\left[-\left(J-\frac{1}{2}\right), -\left(K-\frac{1}{2}\right)\right] & \dots & \dots & h\left[0, -\left(K-\frac{1}{2}\right)\right] & \dots & \dots & h\left[\left(J-\frac{1}{2}\right), -\left(K-\frac{1}{2}\right)\right] \\ \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ \vdots & \dots & h[-1, -1] & h[0, -1] & h[1, -1] & \dots & \vdots \\ h\left[-\left(J-\frac{1}{2}\right), 0\right] & \dots & h[-1, 0] & h[0, 0] & h[1, 0] & \dots & h\left[\left(J-\frac{1}{2}\right), 0\right] \\ \vdots & \dots & h[-1, +1] & h[0, 1] & h[1, +1] & \dots & \vdots \\ \vdots & \dots & \vdots & \vdots & \vdots & \ddots & \vdots \\ h\left[-\left(J-\frac{1}{2}\right), \left(K-\frac{1}{2}\right)\right] & \dots & \dots & h\left[0, \left(K-\frac{1}{2}\right)\right] & \dots & \dots & h\left[\left(J-\frac{1}{2}\right), \left(K-\frac{1}{2}\right)\right] \end{bmatrix}$$

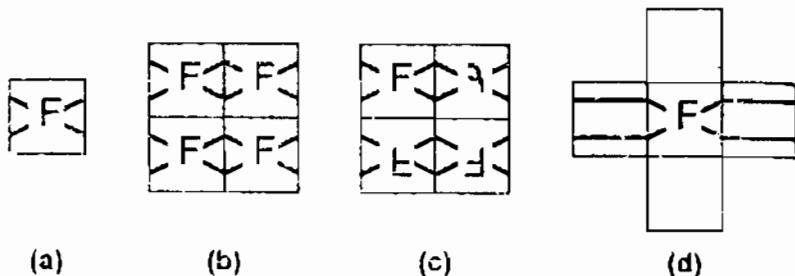
Hình 25: Hệ tọa độ dùng để mô tả $h[j, k]$

Bằng cách đánh giá công thức (84) với $m = n = 0$ trong khi viết lại các giới hạn của tổng chập dựa trên “tâm” của $h[j, k]$, chúng ta thấy rằng các giá trị cần có của $a[j, k]$ lại nằm ngoài các biên ảnh:

$$c[0, 0] = \sum_{j=-J_0}^{+J_0} \sum_{k=-K_0}^{+K_0} h[j, k] a[-j, -k] J_0 = \frac{(J-1)}{2}, K_0 \frac{(K-1)}{2} \quad (87)$$

Câu hỏi đặt ra ở đây là – chúng ta nên gán cho ảnh $a[m, n]$ ($m < 0, m \geq M, n < 0, n \geq N$) những giá trị nào? Câu trả lời cho câu hỏi này là không biết, vì có rất nhiều giá trị khác nhau để chọn lựa, nhưng chúng ta phải biết được các kết cuộc có thể có ứng với chọn lựa của mình. Các phương pháp gán khác nhau tiêu biểu là (a) mở rộng các ảnh bằng một giá trị

độ sáng không đổi (có thể là 0); (b) mở rộng ảnh một cách tuần hoàn; (c) mở rộng ảnh bằng cách lấy đối xứng nó tại các biên; hoặc là (d) mở rộng một cách vô hạn các giá trị tại các biên. Những phương pháp này được minh họa trong hình 26.



Hình 26. Các ví dụ cho các phương pháp mở rộng ảnh ra khỏi các biên hình thức của nó.

Khi tổng chập được viết lại dưới dạng chuẩn theo phương trình (35) cho ảnh $a[m, n]$ (kích thước $M \times N$):

$$c[m, n] = \sum_{j=0}^{M-1} \sum_{k=0}^{N-1} a[j, k] h[m-j, n-k] \quad (88)$$

chúng ta thấy rằng nhân chập $h[j, k]$ đối xứng quanh $j = k = 0$ để tạo thành $h[-j, -k]$ trước khi bị tịnh tiến bởi $[m, n]$ như đã được chỉ ra trong phương trình (88). Trong khi có nhiều nhân chập phổ biến có tính đối xứng theo nghĩa này ($h[j, k] = h[-j, -k]$), một số nhân khác lại không có (xem mục 6.11). Vì vậy chúng ta phải cẩn thận khi cài đặt các bộ lọc có những yêu cầu về đối xứng.

Độ phức tạp tính toán cho một nhân chập kích thước $K \times K$ được cài đặt trong miền tọa độ trên một ảnh $N \times N$ là $O(K^2)$. Ở đây độ phức tạp được đo trên mỗi điểm ảnh dựa trên số các phép nhân và cộng (MADD, hay Multiplies – and – ADDs).

Giá trị tính được bởi một phép chập bắt đầu bằng các giá trị độ sáng nguyên cho $a[m, n]$ có thể cho ra kết quả là số hữu tỷ hoặc số thực trong $c[m, n]$. Do vậy nếu chúng ta chỉ làm việc trên các giá trị độ sáng nguyên, sẽ có sai số làm tròn xuất hiện trong trường hợp này.

Sự khảo sát phương trình (84) bộc lộ một khả năng khác trong việc cài đặt hiệu quả phép chập. Nếu nhân chập $h[j, k]$ là tách được, tức là nếu nó có thể được viết lại dưới dạng:

$$h[j, k] = h_{\text{row}}[k] \cdot h_{\text{col}}[j] \quad (89)$$

thì phép lọc có thể thực hiện theo cách sau:

$$c[m, n] = \sum_{j=0}^{J-1} \left\{ \sum_{k=0}^{K-1} h_{\text{row}}[k] a[m-j, n-k] \right\} h_{\text{col}}[j] \quad (90)$$

Điều này có nghĩa là thay vì áp dụng một bộ lọc 2-chiều, chúng ta có thể áp dụng hai bộ lọc 1-chiều, trước hết theo hướng k và sau đó theo hướng j . Với một ảnh $N \times N$, cách làm này sẽ làm giảm độ phức tạp tính toán trên mỗi điểm ảnh từ $O(J * K)$ xuống còn $O(J + K)$.

Còn có một cách khác để biểu diễn tính tách được nếu chúng ta để ý rằng nhân chập (hình 24) là ma trận h , và nếu tách được thì h có thể viết lại thành:

$$\begin{aligned} [h] &= [h_{\text{col}}] \cdot [h_{\text{row}}]^t \\ (J \times K) &= (J \times 1) \cdot (1 \times K) \end{aligned} \quad (91)$$

trong đó t biểu thị phép chuyển vị ma trận. Nói cách khác, h có thể được biểu diễn dưới dạng tích ngoài của một vector cột $[h_{\text{col}}]$ và một vector hàng $[h_{\text{row}}]$.

Với một số bộ lọc, chúng ta có thể tìm *cài đặt từng bước* cho một phép chập. Khi cửa sổ chập di chuyển trên ảnh (xem phương trình (88)), cột bên trái nhất của dữ liệu ảnh ở phía dưới cửa sổ sẽ bị dịch ra ngoài khi có một cột dữ liệu ảnh mới được dịch vào từ bên phải. Các thuật toán hiệu quả sẽ tận dụng yếu tố này [24] để khi kết hợp với các bộ lọc tách được như đã mô tả ở trên cho ra độ phức tạp tính toán trên mỗi điểm ảnh là $O(\text{constant})$.

6.9.3. Phép chập trong miền tần số

Trong mục 6.4 chúng ta đã chỉ ra rằng có một phương pháp khác để cài đặt phép lọc ảnh thông qua phép chập. Dựa trên phương trình (56), chúng ta thấy hoàn toàn có thể đạt được kết quả như trong phương trình (84) bằng cách sử dụng dãy các thao tác sau:

i) Tính $A(\Omega, \Psi) = F\{a[m,n]\}$

ii) Nhân $A(\Omega, \Psi)$ với phần *đã tính trước* $H(\Omega, \Psi) = F\{h[m,n]\}$

iii) Tính kết quả $c[m,n] = F^{-1}\{A(\Omega, \Psi) \bullet H(\Omega, \Psi)\}$ (92)

Trong khi “cách chế biến” nói trên “(phương trình (92) dường như phá vỡ được các bài toán có liên quan đến phép chập trực tiếp trong miền tọa độ – hay cụ thể hơn là việc xác định các giá trị ảnh nằm ngoài biên ảnh – thì cách tiếp cận theo miền Fourier, trong thực tế, đơn giản chỉ “giả sử rằng” ảnh được lập lại một cách tuần hoàn bên ngoài các biên của nó như được minh họa trong hình 26(b). Hiện tượng này được gọi là *phép chập vòng*.

Nếu phép chập vòng không thể chấp nhận được thì các khả năng khác được minh họa trong hình 26 cũng có thể được

hiện thực hóa bằng cách nhúng ánh $a[m,n]$ và bộ lọc $H(\Omega, \Psi)$ vào các ma trận lớn hơn, với cơ chế mở rộng ánh $a[m,n]$ được cài đặt tương minh.

Độ phức tạp tính toán trên mỗi điểm ảnh theo tiếp cận Fourier cho một ảnh $N \times N$ và nhân chập $K \times K$ là $O(\log N)$ phép cộng và nhân phức độc lập với K . Ở đây chúng ta giả sử rằng $N > K$ và N thường là một số có thể biểu diễn dưới dạng lũy thừa (xem mục 2.1). Giả sử này cho phép chúng ta sử dụng thuật toán biến đổi Fourier nhanh (FFT) nhằm đạt hiệu quả về mặt tính toán. Điều đáng ngạc nhiên là, cách tính gián tiếp đã mô tả trong phương trình (92) lại có thể nhanh hơn cách tính trực tiếp đã nêu trong phương trình (84). Một cách tổng quát, cách làm này đòi hỏi $K^2 \gg \log N$, còn phạm vi của K và N thỏa cách tiếp cận này sẽ phụ thuộc vào sự cài đặt cụ thể trên từng hệ thống máy và hệ điều hành.

6.10 CÁC PHÉP TOÁN LÀM TRƠN ẢNH

Các thuật toán làm trơn thường được áp dụng để giảm nhiễu, hoặc để chuẩn bị ảnh cho những bước xử lý kế tiếp chẳng hạn như trong quá trình phân đoạn ảnh. Chúng được chia thành hai nhóm thuật toán tuyến tính và phi tuyến, trong đó chỉ có nhóm đầu sử dụng đến các phân tích trong miền Fourier. Chúng còn có thể được phân biệt dựa trên cơ sở được cài đặt theo bộ lọc có giá hình chữ nhật hay hình tròn.

6.10.1. Các bộ lọc tuyến tính

Sau đây là một loạt các thuật toán lọc trơn với những giá đỡ hữu dụng nhất.

Bộ lọc đều

Với bộ lọc này, ảnh đầu ra dựa trên trung bình cục bộ của bộ lọc đầu vào với tất cả mọi giá trị nằm bên trong giá

lọc có cùng trọng số. Trong miền tọa độ liên tục (x,y) bảng 11-T.1 và 4-T.3 cho PSF và hàm chuyển tương ứng với hai trường hợp giá chữ nhật và giá tròn. Với miền tọa độ rời rạc $[m,n]$, các giá trị bộ lọc là các mẫu của trường hợp liên tục. Hình 27 cho chúng ta các ví dụ về trường hợp giá chữ nhật ($J = K = 5$) và giá tròn ($R = 2.5$)

$$h_{\text{rect}}[j, k] = \frac{1}{25} \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \end{bmatrix}$$

(a) Bộ lọc chữ nhật ($J = K = 5$)

$$h_{\text{cier}}[j, k] = \frac{1}{21} \begin{bmatrix} 0 & 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 & 0 \end{bmatrix}$$

(b) Bộ lọc tròn ($R = 2.5$)

Hình 27. Các bộ lọc đều để làm trơn ảnh

Chúng ta cần lưu ý là trong cả hai trường hợp bộ lọc đều được chuẩn hóa sao cho $\sum h[j, k] = 1$. Khi đó nếu ảnh đầu vào $a[m, n]$ là hằng số thì ảnh đầu ra $c[m, n]$ cũng có cùng hằng số đó. Sự canh chỉnh ở đây có thể tìm thấy trong phần tính chất của phép biến đổi Fourier, như đã được mô tả trong

phương trình (58). Như chúng ta có thể quan sát thấy từ bảng 11, cả hai loại bộ lọc này đều có các hàm chuyển có mẫu âm, do vậy có thể dẫn đến sự đảo ngược pha như trong hình 24. Cài đặt dạng vuông của bộ lọc có tính tách được và từng bước, còn cài đặt dạng tròn chỉ có tính từng bước (xem [24][25]).

Bộ lọc tam giác

Với bộ lọc này, ảnh đầu ra dựa trên trung bình cục bộ của bộ lọc đầu vào với các giá trị nằm bên trong giá lọc có trọng số khác nhau. Một cách tổng quát, bộ lọc có thể được xem như phép chập của hai bộ lọc đều đồng nhất, tức là cả hai đều có dạng chữ nhật hoặc cả hai đều có dạng tròn. Ý tưởng này có ảnh hưởng trực tiếp đến độ phức tạp tính toán (xem [24], [25] và bảng 13). Trong miền tọa độ liên tục, bảng 11-T.2 và 4-T.4 cho PSF và hàm chuyển tương ứng với hai trường hợp giá chữ nhật và giá tròn. Như chúng ta có thể quan sát thấy từ bảng 11, cả hai loại bộ lọc này đều có các hàm chuyển không có mẫu âm, do vậy không dẫn đến hiện tượng đảo ngược pha.

Hình 28 cho chúng ta các ví dụ về trường hợp giá chữ nhật ($J = K = 5$) và giá tròn ($R = 2.5$). Bộ lọc một lần nữa được chuẩn hóa sao cho $\sum h[j, k] = 1$.

$$h_{\text{rect}}[j, k] = \frac{1}{81} \begin{bmatrix} 1 & 2 & 3 & 2 & 1 \\ 2 & 4 & 6 & 4 & 2 \\ 3 & 6 & 9 & 6 & 3 \\ 2 & 4 & 6 & 4 & 2 \\ 1 & 2 & 3 & 2 & 1 \end{bmatrix}$$

(a) Bộ lọc dạng chóp ($J = K = 5$)

$$h_{\text{circ}}[j, k] = \frac{1}{25} \begin{bmatrix} 0 & 0 & 1 & 0 & 0 \\ 0 & 2 & 2 & 2 & 0 \\ 1 & 2 & 5 & 2 & 1 \\ 0 & 2 & 2 & 2 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix}$$

(b) Bộ lọc dạng nón ($R = 2.5$)

Hình 28. Các bộ lọc tam giác để làm trơn ảnh

Bộ lọc Gauss

Việc sử dụng nhân Gauss để làm trơn là một việc làm hết sức phổ biến. Cách làm này phải được thực hiện theo một số tính chất của bộ lọc Gauss (ví dụ như định lý giới hạn trung tâm, tích không gian – độ rộng giải tần cực tiểu) cùng với nhiều lĩnh vực ứng dụng như dò cạnh và phân tích không gian phạm vi. PSF là hàm chuyển cho bộ lọc Gauss trên không gian liên tục được cho trong bảng 11–T6. Bộ lọc Gauss có tính tách được theo phương trình (93).

$$\begin{aligned} h(x, y) = g_{2D}(x, y) &= \left(\frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{x^2}{2\sigma^2}} \right) \cdot \left(\frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{y^2}{2\sigma^2}} \right) \\ &= g_{1D}(x) \cdot g_{1D}(y) \end{aligned} \quad (93)$$

Sau đây chúng ta sẽ lần lượt khảo sát bốn cách khác nhau để cài đặt bộ lọc Gauss.

Cách 1

Thực hiện phép chập bằng cách sử dụng một số hữu hạn mẫu (N_0) cho bộ lọc Gauss giữ vai trò là nhân chập. Người ta thường chọn $N_0 = [3\sigma]$ hoặc $[5\sigma]$.

$$g_{1D}[n] = \begin{cases} \frac{1}{\sqrt{2\pi\sigma}} e^{-\frac{n^2}{2\sigma^2}} & |n| \leq N_0 \\ 0 & |n| > N_0 \end{cases} \quad (94)$$

Cách 2

Lặp đi lặp lại phép chập với nhân chập là một bộ lọc đều.

$$g_{1D}[n] \approx u[n] \otimes u[n] \otimes u[n]$$

$$u[n] = \begin{cases} \frac{1}{2N_0 + 1} & |n| \leq N_0 \\ 0 & |n| > N_0 \end{cases} \quad (95)$$

Cách cài đặt thật sự (ở mỗi chiều) thường có dạng sau:

$$c[n] = ((a[n] \otimes u[n]) \otimes u[n]) \otimes u[n] \quad (96)$$

Cách cài đặt này sử dụng phép xấp xỉ do định lý giới hạn trung tâm đưa ra. Với giá trị mong muốn σ , trong phương trình (96) chúng ta sử dụng $N_0 = \lceil \sigma \rceil$ mặc dù cách làm này giới hạn đáng kể sự chọn lựa giá trị – trên tập các số nguyên.

Cách 3

Thực hiện phép nhân trong miền tần số. Do biến đổi Fourier một bộ lọc Gauss là một bộ lọc Gauss (xem bảng – T.6), chúng ta chỉ cần chuẩn bị một bộ lọc $H(\Omega, \Psi) = G_{2D}(\Omega, \Psi)$ để sử dụng trong phương trình (92). Để tránh những tác động cắt số trên miền tần số do kích thước vô hạn của bộ lọc Gauss, chúng ta cần phải chọn σ đủ lớn. Thông thường việc chọn $\sigma > k/\pi$ với $k = 3$ hay 4 là hiệu quả.

Cách 4

Sử dụng cách cài đặt bộ lọc đệ qui. Một bộ lọc đệ qui sẽ có đáp ứng xung vô hạn, do vậy có giá vô hạn. Bộ lọc Gauss

tách được cũng có thể được cài đặt bằng cách áp dụng kỹ thuật sau đây ở mỗi chiều khi $\sigma = 0.5$.

i) Chọn σ dựa trên mục đích của việc lọc.

ii) Xác định tham số q dựa trên phương trình (98).

iii) Sử dụng phương trình (99) để xác định các hệ số lọc $\{b_0, b_1, b_2, b_3, B\}$

iv) Áp dụng phương trình sai phân tiến (100).

v) Áp dụng phương trình sai phân lùi (101). (97)

Mối quan hệ giữa giá trị σ và q được cho bởi công thức:

$$q = \begin{cases} 0.98711\sigma - 0.96330 & \sigma \geq 2.5 \\ 3.97156 - 4.14554\sqrt{1 - 0.26891\sigma} & 0.5 \leq \sigma \leq 2.5 \end{cases} \quad (98)$$

Các hệ số lọc $\{b_0, b_1, b_2, b_3, B\}$ được xác định bởi:

$$b_0 = 1.57825 + (2.44413q) + (1.4281q^2) + (0.422205q^3)$$

$$b_1 = (2.44413q) + (1.4281q^2) + (0.422205q^3)$$

$$b_2 = (1.4281q^2) + (0.422205q^3)$$

$$b_3 = 0.422205q^3$$

$$B = 1 - (b_1 + b_2 + b_3) / b_0$$

Phương trình sai phân tiến 1-chiều nhận một hàng (hay cột) đầu vào $a[n]$ và cho kết quả đầu ra trung gian $w[n]$ như sau:

$$w[n] = Ba[n] + (b_1w[n-1] + b_2w[n-2] + b_3 w[n-3])/b_0 \quad (100)$$

Phương trình sai phân lùi 1-chiều nhận kết quả trung gian $w[n]$ và cho ra kết xuất $c[n]$ như sau:

$$c[n] = Bw[n] + (b_1c[n+1] + b_2c[n+2] + b_3 c[n+3])/b_0 \quad (101)$$

Phương trình tiến được áp dụng với $n = 0$ lên đến $n = N - 1$ trong khi phương trình lùi được áp dụng với $n = N - 1$ xuống đến $n = 0$.

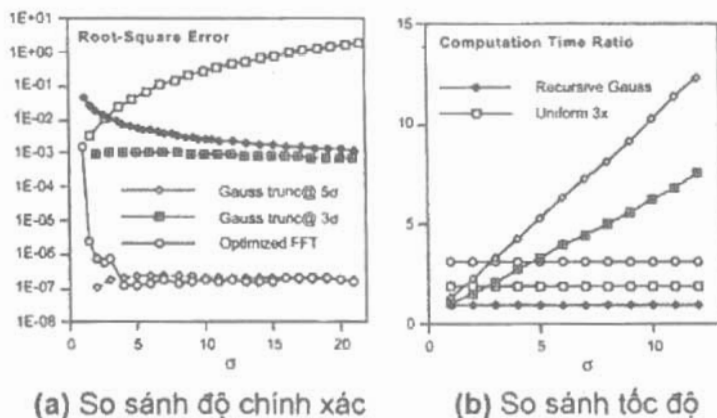
Hiệu năng tương đối của các cách cài đặt bộ lọc Gauss nêu trên có thể mô tả như sau. Bằng cách sử dụng *sai số căn bình phương* $\sqrt{\sum_{n=-x}^{+x} |g[n/\sigma] - h[n]|^2}$ giữa một bộ lọc Gauss thật có kích thước vô hạn $g[n/\sigma]$ và một bộ lọc Gauss xấp xỉ $h[n]$ làm độ đo mức độ chính xác, chúng ta sẽ có các kết quả so sánh như trong hình 29(a). Tốc độ tính toán tương đối của các thuật toán được cho trong hình 29(b).

Độ đo sai số căn bình phương là một độ đo hết sức chặt chẽ, do vậy tất cả các bộ lọc, ngoại trừ bộ lọc đều $3x$ với σ lớn, đều khá chính xác. Cách cài đặt đệ qui là nhanh nhất và độc lập với σ , còn các cài đặt khác chậm hơn một cách đáng kể. Cách cài đặt FFT chẳng hạn, chậm hơn đến 6.1 lần với $N = 256$. Ngoài ra, nó còn đòi hỏi N phải là số có thể biểu diễn được dưới dạng lũy thừa.

Các bộ lọc tron tuyến tính khác

Cách tiếp cận trên miền Fourier mang lại cho chúng ta cơ hội cài đặt được rất nhiều thuật toán làm tron khác nhau. Khi đó các bộ lọc tron sẽ là những bộ lọc thông thấp. Nói chung người ta mong muốn sử dụng một *bộ lọc thông thấp* có pha zero nhằm không tạo ra sự lệch pha khi lọc ảnh. Tầm quan trọng của khái niệm pha được minh họa trong hình 17 và hình 24. Khi những đặc trưng của miền tần số có thể biểu diễn được dưới dạng giải tích, thì chúng ta sẽ có ngay những cách cài đặt tương đối trực tiếp cho $H(\Omega, \mathcal{Y})$. Các ứng cử viên sáng giá ở đây là hai bộ lọc thông thấp "Airy" và

“Exponential Decay” được liệt kê trong bảng 11–T.5 và 4–T.8 tương ứng.



Hình 29. So sánh các thuật toán Gauss khác nhau với $N = 256$. Các ký hiệu được sử dụng cho cả hai đồ thị

6.10.2. Các bộ lọc phi tuyến

Ngoài các bộ lọc tuyến tính, trong việc làm trơn ảnh chúng ta còn có rất nhiều dạng bộ lọc không phải là tuyến tính. Phần tiếp sau đây giới thiệu một số bộ lọc phi tuyến tiêu biểu.

Lọc trung vị

Bộ lọc trung vị được xây dựng trên cơ sở di chuyển một cửa sổ trên ảnh (như trong một phép chập) và tính giá trị điểm ảnh đầu ra dưới dạng giá trị trung bình của các độ sáng thuộc cửa sổ đầu vào. Nếu cửa sổ đó có kích thước $J \times K$ thì chúng ta có thể sắp xếp $J * K$ điểm ảnh theo thứ tự từ giá trị độ sáng nhỏ nhất đến giá trị độ sáng lớn nhất. Nếu $J * K$ là thì giá trị trung vị sẽ là đề mục thứ $(J * K + 1) / 2$ trong danh sách đã được sắp xếp. Chúng ta cần chú ý là giá trị được chọn

phải bằng đúng một trong các giá trị độ sáng đã có sao cho không có lỗi làm tròn xảy ra trong trường hợp chúng ta chỉ muốn làm việc trên các giá trị nguyên. Thuật toán vừa nêu có độ phức tạp ứng với mỗi điểm ảnh là $O(J \cdot K \log(J \cdot K))$. May mắn thay, chúng ta đã có một thuật toán nhanh (do Huang đưa ra trong [23]) cho phép giảm độ phức tạp trên xuống còn $O(K)$ với giả sử là $J \geq K$.

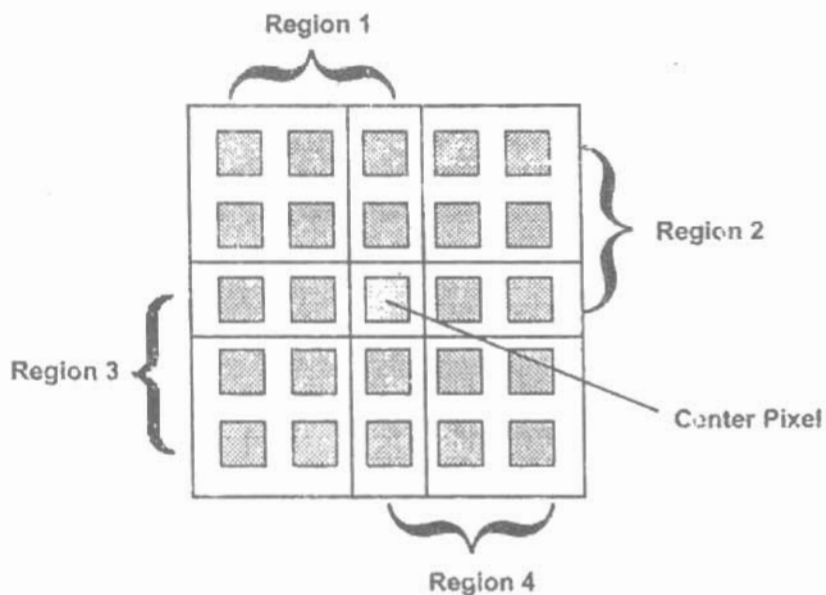
Một biến thể có ích khác của lọc trung vị là *lọc phân trăm*. Ở đây điểm ảnh trung tâm trong cửa sổ được thay thế không phải bởi 50% giá trị sáng mà là $p\%$, trong đó $p\%$ đi từ 0% (*lọc cực tiểu*) đến 100% (*lọc cực đại*). Các giá trị khác với 50% nói chung không tương ứng với các bộ lọc tron.

Lọc Kuwahara

Các cạnh giữ vai trò rất quan trọng trong sự cảm nhận của con người về ảnh (xem hình 16) cũng như trong lĩnh vực phân tích ảnh. Do vậy, việc làm tron ảnh nhưng không làm ảnh hưởng đến độ sắc nét, và nếu có thể, không làm thay đổi vị trí của các cạnh trong ảnh, là hết sức cần thiết. Một bộ lọc thỏa mãn những yêu cầu như vậy gọi là *bộ lọc bảo toàn cạnh*. Một trong những bộ lọc có tính chất như vậy chính là lọc Kuwahara [27] dù bộ lọc này có thể được cài đặt theo nhiều hình dạng cửa sổ khác nhau, thuật toán mà chúng ta mô tả ở đây sử dụng cửa sổ hình vuông có kích thước $J = K = 4L + 1$ trong đó L là số nguyên. Cửa sổ này sẽ được chia thành bốn vùng như trong hình 30.

Trong mỗi vùng i ($i = 1, 2, 3, 4$), chúng ta sẽ đi tính *độ sáng trung bình* m_i theo phương trình (66) và *phương sai* s_i^2 theo phương trình (68). Giá trị đầu ra của điểm ảnh trung

tâm trong cửa sổ chính là giá trị trung bình của vùng có phương sai nhỏ nhất.



Hình 30. Bốn vùng hình vuông cho bộ lọc Kuwahara. Trong ví dụ này $L = 1$ và do vậy $J = K = 5$. Mỗi vùng có kích thước là $[J + \frac{1}{2} \times (K + 1)]/2$.

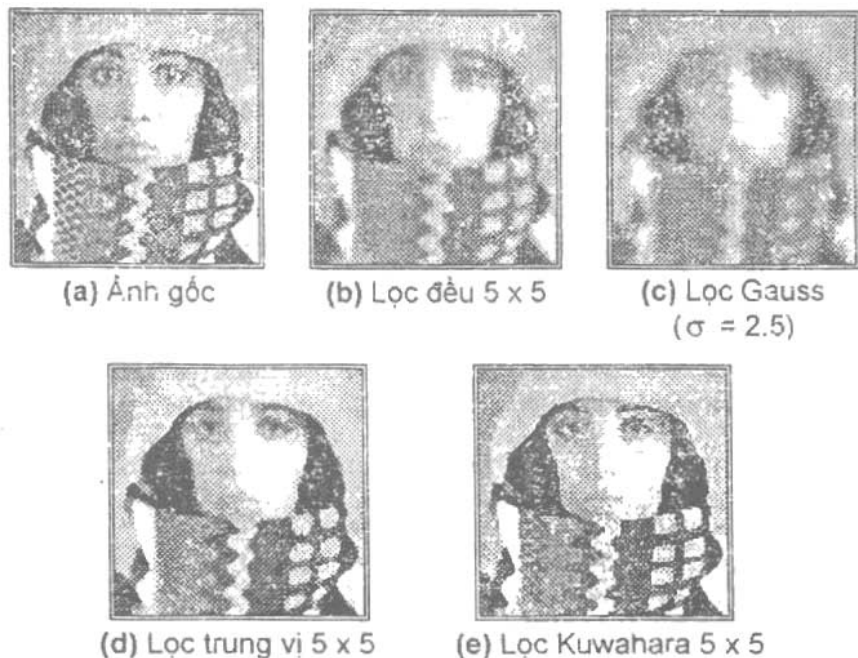
6.10.3. Tóm tắt các thuật toán làm trơn

Bảng sau đây tóm tắt những tính chất của các thuật toán làm trơn đã trình bày ở trên. Kích thước bộ lọc được giả định là bị chặn trong một hình chữ nhật có kích thước $J \times K$ với $J \geq K$. Ảnh có kích thước là $N \times N$.

Bảng 13. Tính chất của các bộ lọc trơn

Thuật toán	Miền	Kiểu	Giá	Tách được/ từng bước	Độ phức tạp/ điểm ảnh
Đều	Không gian	Tuyến tính	Vuông	Có/Có	$O(\text{constant})$
Đều	Không gian	Tuyến tính	Tròn	Không/Có	$O(K)$
Tam giác	Không gian	Tuyến tính	Vuông	Có/không	$O(\text{constant})$
Tam giác	Không gian	Tuyến tính	Tròn	Không/Không	$O(K)$
Gauss	Không gian	Tuyến tính	—	Có/Không	$O(\text{constant})$
Trung vị	Không gian	Tuyến tính	Vuông	Không/Có	$O(K)$
Kuwahara	Không gian	Tuyến tính	Vuông	Không/Không	$O(J \cdot K)$
Khác	Tán số	Tuyến tính	—	—/—	$O(\log N)$

Hình 31 cho chúng ta minh họa về các bộ lọc trơn.



Hình 31. Minh họa các bộ lọc trơn tuyến tính và phi tuyến

6.11. CÁC PHÉP TOÁN ĐẠO HÀM

Tương tự như các phép toán làm trơn ảnh, khả năng lấy đạo hàm theo tọa độ của các điểm ảnh là hết sức quan trọng. Bài toán cơ bản ở đây là, nếu chiếu theo đúng định nghĩa toán học về đạo hàm, chúng ta sẽ không thể thực hiện được việc lấy đạo hàm các điểm ảnh, do một ảnh số hóa không phải là một hàm liên tục $a(x,y)$ theo các biến tọa độ, mà chỉ là một hàm rời rạc $a[m,n]$ với các biến tọa độ nguyên. Vì lý do đó, những thuật toán mà chúng ta trình bày ở đây chỉ có thể được xem là *các xấp xỉ* cho đạo hàm thật sự theo tọa độ của ảnh liên tục ban đầu.

Ngoài ra, như chúng ta có thể quan sát thấy từ những tính chất của phép biến đổi Fourier trong phương trình (59), việc lấy đạo hàm đồng nghĩa với việc nhân phổ tín hiệu với u hoặc v . Điều này có nghĩa là nhiều có tần số cao sẽ được nhấn mạnh trong ảnh đầu ra. Lời giải tổng quát cho bài toán này là kết hợp phép toán đạo hàm với một phép toán khử nhiễu, như phép làm trơn ảnh chẳng hạn.

6.11.1 Các đạo hàm bậc nhất

Trong trường hợp ảnh là một hàm hai biến, chúng ta cần phải xác định các hướng cần lấy đạo hàm. Các hướng có thể có ở đây là hướng dọc, hướng ngang, và hướng tùy ý là sự kết hợp của hai hướng dọc và ngang. Nếu chúng ta sử dụng h_x để ký hiệu bộ lọc đạo hàm theo phương ngang h_y ký hiệu bộ lọc đạo hàm theo phương dọc, h_θ để ký hiệu bộ lọc đạo hàm theo một hướng bất kỳ, thì sẽ có quan hệ sau:

$$[h_\theta] = \cos\theta \cdot [h_x] + \sin\theta \cdot [h_y] \quad (102)$$

Các bộ lọc gradient

Chúng ta cũng có thể tạo ra một mô tả đạo hàm kiểu vectơ dưới dạng *gradient*, ký hiệu là $\nabla a[m,n]$, của một ảnh:

$$\nabla a = \frac{\partial a}{\partial x} \bar{i}_x + \frac{\partial a}{\partial y} \bar{i}_y = (h_x \otimes a) \bar{i}_x + (h_y \otimes a) \bar{i}_y \quad (103)$$

trong đó \bar{i}_x và \bar{i}_y là các vectơ đơn vị theo hai hướng ngang và dọc. Phương trình trên dẫn đến hai mô tả sau đây:

$$\text{Độ lớn gradient } |\nabla a| = \sqrt{(h_x \otimes a)^2 + (h_y \otimes a)^2} \quad (104)$$

$$\text{Hướng gradient } \Psi(\nabla a) = \arctan \left\{ \frac{(h_y \otimes a)}{(h_x \otimes a)} \right\} \quad (105)$$

Độ lớn của gradient đôi khi còn được xấp xỉ bởi:

$$\text{Độ lớn gradient xấp xỉ } |\nabla a| \cong |h_x \otimes a| + |h_y \otimes a| \quad (106)$$

Kết quả sau cùng của những phép toán này phụ thuộc rất nhiều vào cách chọn h_x và h_y . Sau đây chúng ta sẽ xem xét một số cách chọn lựa (h_x, h_y) phổ biến nhất.

Các bộ lọc đạo hàm cơ bản

Các bộ lọc này được đặc tả bởi:

$$\text{i) } [h_x] = [h_y]^t = [1 \ -1] \quad (107)$$

$$\text{ii) } [h_x] = [h_y]^t = [1 \ 0 \ -1]$$

trong đó "t" ký hiệu cho phép chuyển vị ma trận. Hai bộ lọc vừa nêu khác nhau đáng kể về độ lớn Fourier cũng như trong các tính chất pha Fourier. Với phạm vi của tần số đi từ $0 \leq \Omega \leq \pi$ các yếu tố vừa nêu được cho bởi các công thức sau:

$$\text{i) } [h] = [1 \ -1] \xleftrightarrow{F} |H(\Omega)| = 2 \left| \sin \frac{\Omega}{2} \right| \quad \varphi(\Omega) = \frac{\pi - \Omega}{2} \quad (108)$$

$$\text{ii) } [h] = [1 \ 0 \ -1] \xleftrightarrow{F} |H(\Omega)| = 2 |\sin \Omega| \quad \varphi(\Omega) = \frac{\pi}{2}$$

Dạng thứ hai (ii) cho phép loại các số hạng có tần số cao – trong khi dạng (i) không thể. Dạng thứ nhất dẫn đến sự dịch chuyển pha, còn dạng thứ hai thì không.

Các bộ lọc gradient Prewitt

Các bộ lọc này được xác định theo cách sau đây:

$$[h_x] = \frac{1}{3} \begin{bmatrix} 1 & 0 & -1 \\ 1 & 0 & -1 \\ 1 & 0 & -1 \end{bmatrix} = \frac{1}{3} \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} \bullet [1 \ 0 \ -1]$$

$$[h_y] = \frac{1}{3} \begin{bmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ -1 & -1 & -1 \end{bmatrix} = \frac{1}{3} \begin{bmatrix} 1 \\ 0 \\ -1 \end{bmatrix} \bullet [1 \ 1 \ 1] \quad (109)$$

Trong công thức nói trên, cả h_x và h_y đều tách được. Mỗi bộ lọc lấy đạo hàm theo một hướng nhờ phương trình (107) *ii* và làm trơn theo hướng trực giao với hướng đó nhờ một bộ lọc đều 1-chiều, như đã được mô tả trong mục 6.10.1.

Các bộ lọc gradient Sobel

Các bộ lọc này được xác định bởi:

$$[h_x] = \frac{1}{4} \begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{bmatrix} = \frac{1}{4} \begin{bmatrix} 1 \\ 2 \\ 1 \end{bmatrix} \bullet [1 \ 0 \ -1] \quad (110)$$

$$[h_y] = \frac{1}{4} \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix} = \frac{1}{4} \begin{bmatrix} 1 \\ 0 \\ -1 \end{bmatrix} \bullet [1 \ 2 \ 1]$$

Một lần nữa h_x và h_y đều tách được. Mỗi bộ lọc lấy đạo hàm theo một hướng nhờ phương trình (107) *ii* và làm trơn theo hướng trực giao với hướng đó nhờ một bộ lọc tam giác 1-chiều, như đã được mô tả trong mục 6.10.1.

Các bộ lọc gradient khác

Sự có mặt của nhiều kỹ thuật xử lý tín hiệu 1-chiều trong việc thiết kế những bộ lọc số hóa đã mang lại cho chúng ta những công cụ hết sức mạnh mẽ trong việc tạo ra các phiên bản 1-chiều cho h_x và h_y . Bằng cách sử dụng thuật toán thiết kế bộ lọc Parks–McClellan chẳng hạn, chúng ta có thể chọn các dải tần số tại đó cần lấy đạo hàm cũng như các

dải tần tại đó chúng ta muốn loại bỏ nhiễu. Thuật toán này sau đó sẽ cho ra một bộ lọc lẻ, thực, có độ dài cực tiểu và thỏa các đặc tả đã được đưa ra.

Lấy ví dụ như nếu chúng ta muốn một bộ lọc có các đặc trưng đạo hàm trong một dải thông (với trọng số 1.0) với phạm vi tần số đi từ $0.0 \leq \Omega \leq 0.3\pi$ và một dải dừng (với trọng số 3.0) trong phạm vi $0.32\pi \leq \Omega \leq \pi$, thì thuật toán nêu trên sẽ tạo ra một bộ lọc 7 mẫu tối ưu sau:

$$[h_x] = [h_y]^t = \frac{1}{16348} [-3571 \ 8212 \ -15580 \ 0 \ 15580 \ -8212 \ 3571] \quad (111)$$

Gradient sau đó có thể tính được nhờ phương trình (103).

Các bộ lọc gradient Gauss

Trong xử lý ảnh hiện đại, một trong những kỹ thuật phổ biến nhất là sử dụng một bộ lọc Gauss (xem mục 6.10.1) để đạt được một độ trơn ảnh mong muốn và tính được một trong số các đạo hàm được liệt kê trong phương trình (107). Do vậy, trước hết chúng ta có thể áp dụng bộ lọc Gauss đệ qui trong phương trình (97) theo sau bởi phương trình (107)ii để thu được các bộ lọc đạo hàm mong muốn h_x và h_y . Ngoài ra, để đạt được độ hiệu quả cao trong tính toán, chúng ta có thể kết hợp hai bước này lại như sau:

$$w[n] = \left(\frac{B}{2}\right) (a[n+1] - a[n-1]) + (b_1 w[n-1] + b_2 w[n-2] + b_3 w[n-3]) / b_0$$

$$c[n] = Bw[n] + (b_1 c[n+1] + b_2 c[n+2] + b_3 c[n+3]) / b_0 \quad (112)$$

trong đó các hệ số được xác định bởi phương trình (99). Phương trình tiến đầu tiên được áp dụng từ $n = 0$ đến $n = N - 1$, trong khi đó phương trình lùi thứ hai được áp dụng từ $n = N - 1$ đến $n = 0$.

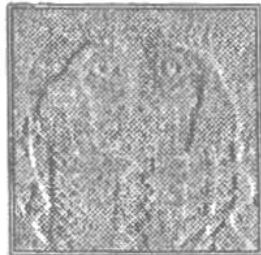
Tóm tắt

Các ví dụ về tác động của *những thuật toán đạo hàm* trên một phiên bản có nhiễu của hình 31(a) (SNR = 29 dB) được thể hiện trong hình 32 (a) – (c). Trong khi đó, kết quả của các *thuật toán gradient độ lớn* áp dụng cho hình 31(a) được thể hiện trong hình 33(a) – (c). Sau khi xử lý, tất cả các ảnh đều được co giãn độ tương phản theo phương trình (77) nhằm giúp cho việc thể hiện trên màn hình rõ ràng hơn.

Gradient độ lớn được lấy trên những giá trị lớn tại các vị trí có cạnh rõ rệt trong ảnh. Sự lựa chọn giá trị – thích hợp trong các phương pháp lấy đạo hàm kiểu bộ lọc Gauss (hình 32(c)) hay gradient (hình 33(c)) cho phép tính bất kỳ các dạng đạo hàm khác như đạo hàm đơn giản, đạo hàm Prewitt, Sobel, ... Theo nghĩa này thì đạo hàm theo kiểu lọc Gauss đại diện cho một tập các bộ lọc đạo hàm.



(a) Đạo hàm đơn giản – phương trình (107)ii



(b) Đạo hàm với lọc Sobel – phương trình (110)



(c) Đạo hàm với lọc Gauss ($\sigma=1.5$) & phương trình (107)ii

Hình 32. Các thuật toán tính đạo hàm ngang h_x



(a) Đạo hàm đơn giản – phương trình (107)ii



(b) Đạo hàm với lọc Sobel – phương trình (110)



(c) Đạo hàm với lọc Gauss ($\sigma=1.5$) & phương trình (107)ii

Hình 33: Các thuật toán tính gradient độ lớn $|\nabla a|$

6.11.2. Các đạo hàm bậc hai

Ngoài các đạo hàm bậc nhất, chúng ta hoàn toàn có thể tính được các đạo hàm bậc cao hơn của một hàm hai biến, trong xử lý ảnh, như chúng ta sẽ có dịp tìm hiểu trong các mục 7.2.1 và 7.3.2, các đạo hàm bậc hai hay toán tử Laplace giữ một vai trò rất quan trọng. Toán tử Laplace được xác định bởi công thức sau:

$$\nabla^2 a = \frac{\partial^2 a}{\partial x^2} + \frac{\partial^2 a}{\partial y^2} = (h_{2x} \otimes a) + (h_{2y} \otimes a) \quad (113)$$

trong đó h_{2x} và h_{2y} là các bộ lọc đạo hàm bậc hai. Trong miền tần số, từ phương trình (59) chúng ta có:

$$\nabla^2 a \leftrightarrow - (u^2 + v^2) A(u, v) \quad (114)$$

Hàm chuyển của một toán tử Laplace tương ứng với một parabol $(u, v) = - (u^2 + v^2)$.

Bộ lọc đạo hàm bậc hai cơ bản

Bộ lọc này được xác định bởi:

$$[h_{2x}] = [h_{2y}]t = [1 \quad -2 \quad 1] \quad (115)$$

Phổ tần số theo mỗi hướng của nó sẽ là:

$$H(\Omega) = F\{1 \quad -2 \quad 1\} = -2(1 - \cos\Omega) \quad (116)$$

với $-\pi \leq \Omega \leq +\pi$. Ở đây chúng ta có thể sử dụng hai bộ lọc 1-chiều theo cách thức đã được chỉ ra trong phương trình (113), hoặc kết hợp cả hai thành một bộ lọc 2-chiều như sau:

$$[h] = \begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix} \quad (117)$$

Sau đó phương trình (84) sẽ được sử dụng để cho ra kết quả mong muốn.

Toán tử Laplace trên miền tần số

Bộ lọc này chính là sự cài đặt của phương pháp đã được nói đến trong phương trình (92). Nó có dạng sau đây:

$$c[m,n] = F^{-1} \left\{ -(\Omega^2 + \Psi^2) A(\Omega, \Psi) \right\} \quad (118)$$

Bộ lọc đạo hàm bậc hai kiểu Gauss

Đây là sự mở rộng trực tiếp từ bộ lọc đạo hàm bậc nhất theo kiểu Gauss đã mô tả ở trên, và có thể áp dụng độc lập trong mỗi chiều. Trước tiên chúng ta áp dụng phép toán làm trơn kiểu Gauss với giá trị σ được chọn theo yêu cầu của từng bài toán cụ thể. Sau đó bộ lọc đạo hàm bậc hai theo phương trình (115) hoặc (118) sẽ được áp dụng. Một lần nữa chúng ta sẽ có rất nhiều thuật toán làm trơn kiểu Gauss để lựa chọn trong tình huống này.

Nhằm đạt được sự hiệu quả trong tính toán, chúng ta có thể tận dụng cách cài đặt đệ qui và kết hợp hai bước – làm trơn và lấy đạo hàm – theo cách sau đây:

$$\begin{aligned} w[n] &= B(a[n] - a[n-1]) + (b_1 w[n-1] + b_2 w[n-2] + b_3 w[n-3]) / b_0 \\ c[n] &= B(w[n+1] - w[n]) + (b_1 c[n+1] + b_2 c[n+2] + b_3 c[n+3]) / b_0 \end{aligned} \quad (119)$$

Với các hệ số được xác định nhờ phương trình (99). Phương trình tiên đầu tiên được áp dụng với $n = 0$ đến $n = N - 1$, còn phương trình lùi thứ hai được áp dụng với $n = N - 1$ đến $n = 0$.

Các bộ lọc Laplace khác

Một lần nữa, các kỹ thuật thiết kế những bộ lọc số hóa lại mang lại cho chúng ta những công cụ hết sức mạnh mẽ để tạo ra những bộ lọc được tối ưu hóa cho từng bài toán cụ thể. Bằng cách sử dụng thuật toán thiết kế bộ lọc Parks–McClellan chẳng hạn, chúng ta có thể chọn các dải tần số tại đó cần lấy đạo hàm cũng như các dải tần tại đó chúng ta muốn loại bỏ nhiều. Thuật toán này sau đó sẽ cho ra một bộ lọc chắn, thực, có độ cài cực tiểu và thỏa các đặc tả đã được đưa ra.

Lấy ví dụ như nếu chúng ta muốn một bộ lọc có các đặc trưng đạo hàm bậc hai trong một dải thông (với trọng số 1.0) với phạm vi tần số đi từ $0.0 \leq \Omega \leq 0.3\pi$ và một dải dừng (với trọng số 3.0) trong phạm vi $0.32\pi \leq \Omega \leq \pi$, thì thuật toán nêu trên sẽ tạo ra một bộ lọc 7 mẫu tối ưu sau:

$$[h_x] = [h_y]^t = \frac{1}{11043} \{-3448 \ 10145 \ 1495 \ -16383 \ 1495 \ 10145 \ -3448\} \quad (120)$$

Sau đó toán tử Laplace có thể được tính theo phương trình (113).

Lọc SDGD

Có một bộ lọc hết sức hữu ích cho việc dò cạnh và đo đặc các đối tượng là bộ lọc đạo hàm bậc hai theo hướng gradient (SDGD, hay *Second-Derivative-in-the-Gradient-Direction*). Bộ lọc này sử dụng năm đạo hàm riêng phần sau:

$$\begin{aligned} A_{xx} &= \frac{\partial^2 a}{\partial x^2}, & A_{xy} &= \frac{\partial^2 a}{\partial x \partial y}, & A_x &= \frac{\partial a}{\partial x} \\ A_{yx} &= \frac{\partial^2 a}{\partial x \partial y}, & A_{yy} &= \frac{\partial^2 a}{\partial y^2}, & A_y &= \frac{\partial a}{\partial y} \end{aligned} \quad (121)$$

Chúng ta cần chú ý rằng $A_{xy} = A_{yx}$, do vậy chỉ có đúng năm đạo hàm phân biệt.

Bộ lọc SDGD này kết hợp các đạo hàm riêng phần khác nhau theo cách sau:

$$\text{SDGD}(a) = \frac{A_{xx}A_x^2 + 2A_{xy}A_xA_y + A_{yy}A_y^2}{A_x^2 + A_y^2} \quad (122)$$

Như chúng ta có thể quan sát, sự có mặt của một lượng lớn các đạo hàm trong bộ lọc này cho thấy việc khử nhiễu là rất quan trọng, và các bộ lọc đạo hàm theo kiểu Gauss – cả bậc nhất và bậc hai – đều là những ứng viên sáng giá cho loại yêu cầu này [28]. Chúng ta cũng chú ý rằng các bộ lọc đạo hàm bậc nhất và bậc hai đều có cùng dải thông và dải dừng. Điều này có nghĩa là nếu bộ lọc đạo hàm bậc nhất h_{1x} được cho bởi [10 - 1] (phương trình (107)ii) thì bộ lọc đạo hàm bậc hai nên được cho bởi $h_{1x} \otimes h_{1x} = h_{2x} [1 \ 0 \ -2 \ 0 \ 1]$.

Tóm tắt

Kết quả thực hiện của các bộ lọc đạo hàm bậc hai đã trình bày ở trên được minh họa trong hình 34 (a) – (3). Tất cả các ảnh đều đã được co giãn độ tương phản để hiển thị tốt hơn trên màn hình bằng cách sử dụng phương trình (78) và các tham số 1% và 99%.



(a) Toán tử Laplace – phương trình (117)



(b) Parabol Fourier - phương trình (118)



(c) Bộ lọc Gauss ($\sigma=1.0$) – phương trình (117)



(d) "Nhà thiết kế" – phương trình (120)

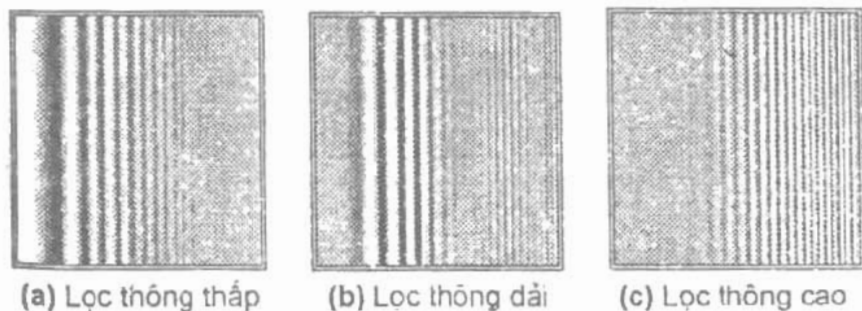


(e) SDGD ($\sigma=1.0$) – phương trình (122)

Hình 34: Các thuật toán có liên quan đến bộ lọc Laplace

6.11.3. Các bộ lọc khác

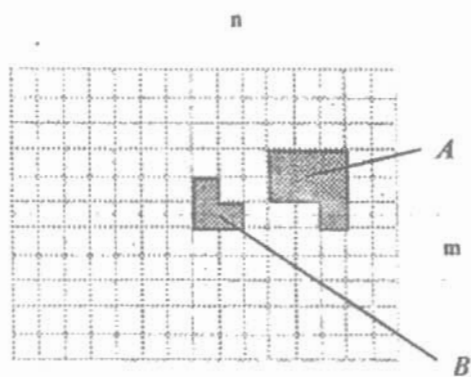
Có vô số các bộ lọc tuyến tính và phi tuyến có thể sử dụng cho xử lý ảnh, do vậy trong mục này chúng ta chỉ có thể trình bày những kiểu cơ bản nhất. Độc giả nào muốn tìm hiểu sâu hơn về chủ đề này có thể tìm trong phần tài liệu tham khảo những thông tin chi tiết về rất nhiều bộ lọc khác (xem phần “Tài liệu tham khảo”). Để hiểu rõ tác động của một bộ lọc hay một lớp các bộ lọc, điều quan trọng là chúng ta chỉ nên sử dụng một tập hợp nhỏ và nhất quán các ảnh kiểm tra phù hợp với một lĩnh vực ứng dụng cụ thể. Tác động của các bộ lọc trên ảnh thường có thể hiểu được dựa vào việc sử dụng ảnh có nhiều vùng với kích thước khác nhau nhằm trực quan hóa tác động xảy ra đối với cạnh, hoặc bằng cách sử dụng những mẫu kiểm tra như các chỗi hình sin để trực quan hóa tác động xảy ra trên miền tần số. Phương pháp đầu đã được sử dụng ở trên trong các hình 22, 24 và 31–34, còn phương pháp thứ hai được minh họa dưới đây trong hình 35.



Hình 35. Các thuật toán chập khác nhau được áp dụng cho ảnh kiểm tra hình sin

6.12. CÁC PHÉP TOÁN MORPHOLOGY

Trong chương 2 chúng ta đã định nghĩa ảnh là một hàm hai biến thực $a(x,y)$ hoặc một mảng các giá trị rời rạc $a[m,n]$. Chúng ta còn có một định nghĩa khác nữa về ảnh, dựa trên quan sát là mỗi ảnh có thể xem như một tập hợp các tọa độ rời rạc hoặc liên tục. Theo một nghĩa nào đó, tập hợp này tương ứng với các điểm ảnh thuộc về các đối tượng hiện hữu trong ảnh. Hình 36 minh họa ý tưởng này bằng cách cho thấy có hai đối tượng, hay hai tập A và B , trong ảnh. Ở đây chúng ta cần phải xác định rõ ràng hệ trục tọa độ như trong hình vẽ. Trong phạm vi giáo trình này, chúng ta chỉ quan tâm đến các giá trị điểm ảnh nhị phân, như đã trình bày trong mục 2.1 và 6.8.1, và giới hạn không gian làm việc trên không gian rời rạc (Z^2). Độc giả có thể tìm thấy những nội dung tổng quát hơn có liên quan đến vấn đề này trong các tài liệu [6,7,29].



Hình 36. Một ảnh nhị phân gồm hai tập đối tượng A và B

Đối tượng A gồm những điểm ảnh α có cùng tính chất sau:

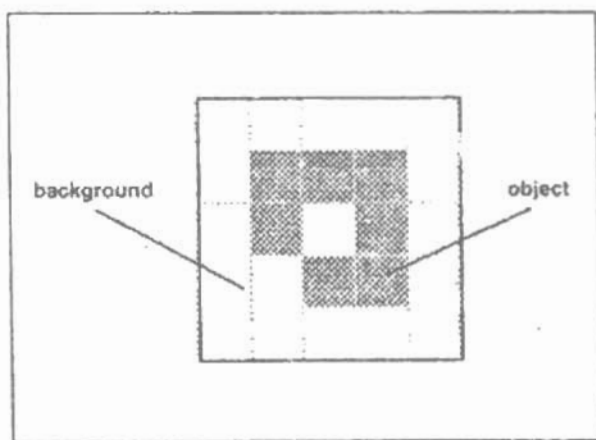
$$\text{Đối tượng} - A = \{\alpha \mid \text{property}(\alpha) == \text{TRUE}\} \quad (123)$$

Lấy ví dụ, đối tượng B trong hình 36 bao gồm các điểm ảnh $[[0,0], [1,0], 0,1]]$.

Phần nền của A , được ký hiệu bởi A^c (phần bù của A), sẽ bao gồm các điểm ảnh không thuộc A :

$$\text{Phần nền} - A^c \{ \alpha \mid \alpha \notin A \} \quad (124)$$

Hẳn độc giả còn nhớ là trong hình 4 chúng ta đã có giới thiệu về tính liên thông của một lân cận. Ở đây chúng ta có thể quan sát thấy là nếu một đối tượng A được định nghĩa dựa trên một lân cận liên thông C ($C = 4, 6$ hoặc 8) thì phần nền A^c tương ứng sẽ có tính liên thông cho bởi $12 - C$. Nguyên nhân dẫn đến sự cần thiết phải có định nghĩa rõ ràng về tính liên thông qua đối tượng và phần nền tương ứng với nó được chỉ ra trong hình 37. Hình này sử dụng hệ trục tọa độ Descartes thông thường.



Hình 37. Mỗi ảnh nhị phân đòi hỏi phải có định nghĩa cẩn thận về tính liên thông của đối tượng trong ảnh và tính liên thông của nền ảnh

6.12.1 Các định nghĩa cơ bản

Các phép toán cơ bản ứng với một đối tượng chính là tập các phép toán tập hợp chuẩn bao gồm phép hội, phép giao, phép lấy bù $\{\cup, \cap, c\}$ và phép tịnh tiến:

* *Phép tịnh tiến* – Cho trước một vector x và một tập hợp A , phép tịnh tiến $A + x$ được định nghĩa như sau:

$$A + x = \{\alpha + x \mid \alpha \in A\} \quad (125)$$

Chúng ta cần lưu ý rằng, do ảnh số hóa chỉ bao gồm các điểm ảnh có tọa độ nguyên trên không gian (Z^2), định nghĩa trên đây sẽ giới hạn số các vector tịnh tiến có thể có x .

Sau đây chúng ta sẽ định nghĩa hai phép toán tập hợp cơ bản, gọi là *các phép toán Minkowski* – bao gồm phép cộng và phép trừ. Trước hết cần chú ý rằng những phần tử riêng lẻ hợp thành B không chỉ có các điểm ảnh mà còn có cả *những vector* khi chúng có vị trí tọa độ xác định so với điểm gốc $[0,0]$. Cho trước hai tập hợp A và B , chúng ta có:

$$\text{Phép cộng Minkowski} - A \oplus B = \bigcup_{\beta \in B} (A + \beta) \quad (126)$$

$$\text{Phép trừ Minkowski} - A \ominus B = \bigcap_{\beta \in B} (A + \beta) \quad (127)$$

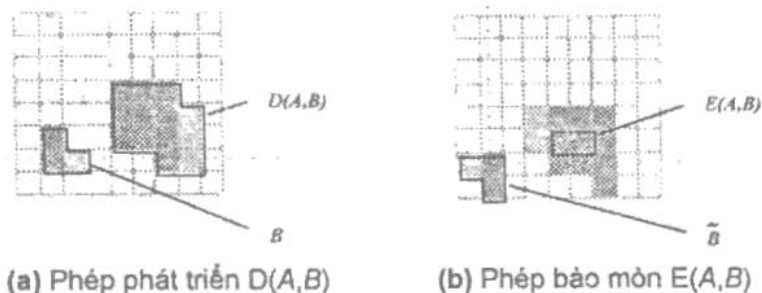
6.12.2 Phép phát triển và phép bào mòn

Từ hai phép toán Minkowski vừa định nghĩa ở trên, chúng ta có thể định nghĩa hai phép toán cho morphology cơ bản là *phép phát triển* và *phép bào mòn*.

$$\text{Phép phát triển} - D(A, B) = A \oplus B \bigcup_{\beta \in B} (A + \beta) \quad (128)$$

$$\text{Phép bào mòn} - E(A, B) = A \ominus B \bigcap_{\beta \in B} (A - \beta) \quad (129)$$

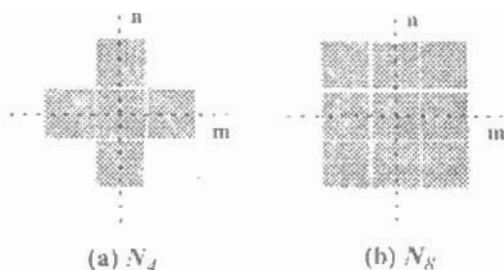
trong đó $\bar{B} = \{-\beta | \beta \in B\}$. Hình 38 minh họa hai phép toán này cho các đối tượng đã được chỉ ra trong hình 36.



Hình 38. Một ảnh nhị phân gồm hai tập đối tượng A và B . Ba điểm ảnh trong B được “mã hóa màu” cho thấy kết quả tương ứng sau khi biến đổi

Mặc dù cả A hay B đều có thể được xem là “ảnh”, nhưng thông thường A được xem là ảnh, còn B được gọi là phần tử cấu trúc. Trong lý thuyết toán học về morphology, phần tử cấu trúc có vai trò tương tự như khái niệm nhân chập trong lý thuyết về các bộ lọc tuyến tính.

Một cách tổng quát thì *phép phát triển* làm cho các đối tượng gia tăng kích thước, còn *phép bào mòn* sẽ làm cho chúng co lại. Sự gia tăng hoặc co lại nhiều hay ít, gia tăng hoặc co lại theo cách thức nào của các đối tượng đều phụ thuộc vào cách lựa chọn phần tử cấu trúc. Hai phần tử cấu trúc thông dụng nhất (sử dụng hệ tọa độ Descartes) là các tập liên thông 4 và liên thông 8, thường được ký hiệu là N_4 và N_8 . Chúng được minh họa trên hình 39.



Hình 39: Các phần tử cấu trúc chuẩn N_4 và N_8 .

Phép phát triển và phép bào mòn có các tính chất sau:

$$\text{Tính giao hoán} - D(A,B) = A \oplus B = B \oplus A = D(B,A) \quad (130)$$

$$\text{Tính không giao hoán} - E(A, B) \neq E(B,A) \quad (131)$$

$$\text{Tính kết hợp} - A \oplus (B \oplus C) = (A \oplus B) \oplus C \quad (132)$$

$$\text{Tính bất biến trong tịnh tiến} - A \oplus (B + x) = (A \oplus B) + x \quad (133)$$

$$\begin{aligned} \text{Tính đối ngẫu} \quad D^c(A, B) &= E(A^c, B^c) \\ E^c(A, B) &= D(A^c, B^c) \end{aligned} \quad (134)$$

Với A là một đối tượng và A^c là phần nền của nó, phương trình (134) cho thấy *phép phát triển* một đối tượng tương đương với *phép bào mòn* phần nền của nó. Tương tự như vậy, *phép bào mòn* một đối tượng tương đương với *phép phát triển* phần nền của nó.

Ngoại trừ một số trường hợp đặc biệt sau:

$$\text{Tính không khả nghịch} - D(E(A,B), B) \neq A \neq E(D(A,B), B) \quad (135)$$

Phép bào mòn có tính chất tịnh tiến sau:

Tính bất biến trong tịnh tiến

$$A e (B + x) = (A + x) e B = (A e B) + x \quad (136)$$

Phép phát triển và phép bào mòn còn có những tính chất quan trọng sau đây. Với một phần tử cấu trúc tùy ý B và hai đối tượng ảnh A_1 và A_2 sao cho $A_1 \subset A_2$ (A_1 là tập con thật sự của A_2):

$$\begin{aligned} \text{Tính tăng trong } A - & D(A_1, B) \subset D(A_2, B) \\ & E(A_1, B) \subset E(A_2, B) \end{aligned} \quad (137)$$

Với hai phần tử cấu trúc B_1 và B_2 sao cho $B_1 \subset B_2$:

$$\text{Tính giảm trong } B - E(A, B_1) \supset E(A, B_2) \quad (138)$$

Các định lý phân rã được nêu ra dưới đây giúp chúng ta có thể tìm ra những cài đặt có hiệu quả cho các bộ lọc morphology.

Phép phát triển -

$$A \oplus (B \cup C) = (A \oplus B) \cup (A \oplus C) = (B \cup C) \oplus A \quad (139)$$

$$\text{Phép bào mòn} - A e (B \cup C) = (A e B) \cap (A e C) \quad (140)$$

$$\text{Phép bào mòn} - (A e B) e C = A e (B \oplus C) \quad (141)$$

$$\text{Đa phát triển} - nB = \underbrace{(B \oplus B \oplus B \oplus \dots \oplus B)}_{n \text{ lần}} \quad (142)$$

Chúng ta còn có một định lý phân rã rất quan trọng do Vincent đưa ra [30]. Trước khi trình bày nội dung chính của định lý này, chúng ta cần phải nêu thêm một số định nghĩa. Một tập lồi trong R^2 là một tập hợp có tính chất sau: mỗi đường thẳng nối hai điểm bất kỳ trong tập này đều chứa các điểm cũng thuộc về tập đó. Khi áp dụng khái niệm này cho một ảnh số, chúng ta cần phải hết sức cẩn trọng với khái

niệm “đường thẳng” trong không gian tọa độ nguyên Z^2 . Ngoài ra, một tập hợp được gọi là *bị chặn* nếu mỗi phần tử của nó đều có độ lớn hữu hạn. Trong trường hợp ảnh số, độ lớn này chính là khoảng cách từ mỗi điểm ảnh đến gốc tọa độ. Một tập hợp được gọi là *đối xứng* nếu $B = -B$. Các tập hợp N_4 và N_8 trong hình 39 là những ví dụ về các tập lồi, tập bị chặn, tập đối xứng.

Định lý do Vincent đưa ra, khi áp dụng cho một ảnh gồm các điểm ảnh rời rạc, khẳng định rằng với một phần tử cấu trúc bị chặn và đối xứng B không chứa các lỗ bên trong, đồng thời chứa tâm của chính nó, $[0,0] \in B$, chúng ta sẽ có kết quả sau đây:

$$D(A,B) = A \oplus B = A \cup (\partial A \oplus B) \quad (143)$$

với ∂A là đường biên của đối tượng. Nói cách khác, ∂A là tập các điểm ảnh có một điểm ảnh nằm trong lân cận. Từ định lý này cho thấy chúng ta không cần phải xử lý tất cả các điểm ảnh thuộc một đối tượng khi thực hiện phép phát triển hay (bằng cách sử dụng phương trình (134)) phép bào mòn, mà chỉ cần xử lý các điểm ảnh biên của nó. Kết quả này còn đúng cho tất cả các phép toán được dẫn xuất từ phép phát triển và phép bào mòn. Bằng cách này, trừ các ảnh bệnh học ra, độ phức tạp tính toán có thể giảm từ $O(N^2)$ xuống còn $O(N)$ cho một ảnh có kích thước $N \times N$. Dựa trên kết quả này, các nhà nghiên cứu đã đưa ra một loạt các thuật toán “nhanh” cho biến đổi morphology, mà chúng ta có thể tìm thấy trong hầu hết các tài liệu chuyên ngành có liên quan đến lĩnh vực này, như trong [30 – 32]. Các thuật toán phát triển và bào mòn đơn giản nhất thường được mô tả như sau đây.

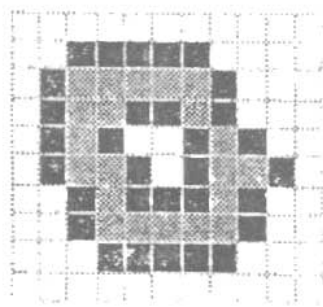
Phép phát triển

Ứng với mỗi điểm ảnh thuộc đối tượng (có trị "1"), thay đổi giá trị của tất cả các điểm ảnh nền (có trị "0") thuộc liên thông C của nó sang giá trị "1".

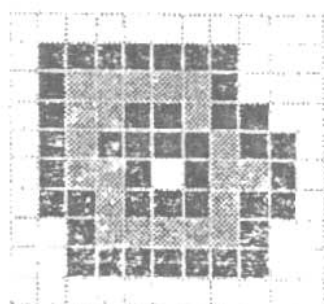
Phép bào mòn

Ứng với mỗi điểm ảnh nền (có trị "0"), thay đổi giá trị của tất cả các điểm ảnh thuộc đối tượng (có trị "1") thuộc liên thông C của nó sang giá trị "0".

Bằng cách so sánh hai thuật toán này với phương trình (143), trong đó $B = N^{C=4}$ hoặc $N^{C=8}$, chúng ta có thể thấy rằng chúng tương đương với các định nghĩa hình thức của hai phép phát triển và bào mòn. Hình 40 minh họa thuật toán cài đặt phép phát triển.



(a) $B = N4$



(b) $B = N8$

Hình 40: Minh họa cho thuật toán phát triển. Các điểm ảnh thuộc đối tượng ban đầu có màu xám, còn những điểm ảnh được thêm vào nhờ thuật toán phát triển có màu đen.

6.12.3. Phép chập logic

Một đối tượng ảnh nhị phân (hay một phần tử cấu trúc) tùy ý A có thể được biểu diễn dưới dạng sau:

$$A \leftrightarrow \sum_{k=-r}^{+r} \sum_{j=-r}^{+r} a[j, k] \bullet \delta[m - j, n - k] \quad (144)$$

trong đó Σ và \bullet là các phép toán logic *OR* và *AND* như đã định nghĩa trong các phương trình (81) và (82), $a[j, k]$ là một hàm đặc trưng nhận các giá trị logic "1" và "0" xác định bởi quan hệ sau:

$$a[j, k] = \begin{cases} 1 & a \in A \\ 0 & a \notin A \end{cases} \quad (145)$$

còn $\delta[m, n]$ là phiên bản logic của hàm delta Dirac nhận các giá trị logic "1" và "0" xác định bởi quan hệ sau:

$$\delta[j, k] = \begin{cases} 1 & j = k = 0 \\ 0 & \text{ngược lại} \end{cases} \quad (146)$$

Khi đó phép phát triển ảnh nhị phân được viết lại thành:

$$D(A, B) \leftrightarrow \sum_{k=-r}^{+r} \sum_{j=-r}^{+r} a[j, k] \bullet b[m - j, n - k] = a \otimes b \quad (147)$$

và do hai phép toán logic *OR* và *AND* có tính giao hoán, chúng ta còn có thể viết lại quan hệ trên dưới dạng:

$$D(A, B) \leftrightarrow \sum_{k=-r}^{+r} \sum_{j=-r}^{+r} a[m - j, n - k] \bullet b[j, k] = b \otimes a = D(B, A) \quad (148)$$

Bằng cách sử dụng định lý De Morgan:

$$\overline{(a + b)} = \bar{a} \bullet \bar{b} \quad \overline{(a \bullet b)} = \bar{a} + \bar{b} \quad (149)$$

Kết hợp với các phương trình (148) và (134), phép bào mòn sẽ được viết lại dưới dạng sau:

$$E(A, B) = \prod_{k=-\infty}^{+\infty} \prod_{j=-\infty}^{+\infty} (a[m - j, n - k] + \bar{b}[j, k]) \quad (150)$$

Như vậy, *phép phát triển* và *phép bào mòn* trên ảnh nhị phân có thể được xem là một dạng của phép chập trên đại số Boole.

Trong mục 6.9.2 chúng ta sẽ có dịp thấy rằng, khi phép chập được sử dụng thì sự lựa chọn các điều kiện biên cho một ảnh là cần thiết. Sự kiện phép phát triển và bào mòn trở thành một phép chập logic không phải là một ngoại lệ. Có hai lựa chọn phổ biến nhất là: hoặc tất cả mọi thứ nằm ngoài ảnh nhị phân đều có trị "0", hoặc tất cả mọi thứ nằm ngoài ảnh nhị phân đều có trị "1".

6.12.4. Phép mở và phép đóng

Chúng ta có thể kết hợp phép phát triển và phép bào mòn để tạo thành các phép toán có mức độ quan trọng cao hơn:

$$\text{Phép mở} - O(A, B) = A \cdot B = D(E(A, B), B) \quad (151)$$

$$\text{Phép đóng} - C(A, B) = A \bullet B = E(D(A, -B) - B) \quad (152)$$

Các phép mở và đóng có các tính chất sau:

$$\begin{aligned} \text{Tính đối ngẫu} - C^c(A, B) &= O(A^c, B) \\ O^c(A, B) &= C(A^c, B) \end{aligned} \quad (153)$$

$$\begin{aligned} \text{Tính tịnh tiến} - O(A + x, B) &= O(A, B) + x \\ C(A + x, B) &= C(A, B) + x \end{aligned} \quad (154)$$

Với *phép toán mở* thông qua phần tử cấu trúc B và các ảnh A, A_1, A_2 trong đó A_1 là ảnh con của A_2 ($A_1 \subseteq A_2$), chúng ta có các tính chất sau:

$$\text{Tính phản mở rộng} - O(A, B) \subseteq A \quad (155)$$

$$\text{Tính đơn điệu tăng} - O(A_1, B) \subseteq O(A_2, B) \quad (156)$$

$$\text{Tính dừng} - O(O(A, B), b) = O(A, B) \quad (157)$$

Với phép toán đóng thông qua phần tử cấu trúc B và các ảnh A, A_1, A_2 , trong đó A_1 là ảnh con của A_2 ($A_1 \subseteq A_2$), chúng ta có:

$$\text{Tính phản mở rộng} - A \subseteq C(A, B) \quad (158)$$

$$\text{Tính đơn điệu tăng} - C(A_1, B) \subseteq C(A_2, B) \quad (159)$$

$$\text{Tính dừng} - C(C(A, B), B) = C(A, B) \quad (160)$$

Hai tính chất quan trọng cho bởi các phương trình (155) và (84) hết sức quan trọng trong lý thuyết toán về morphology, đến độ chúng có thể được xem là nguyên nhân dẫn đến việc định nghĩa phép bào mòn với $-B$, thay vì với B trong phương trình (129).

6.12.5. Phép trùng – trật

Toán tử *trùng – trật* do Serra đưa ra có định nghĩa như sau. Cho trước một ảnh A và hai phần tử cấu trúc B_1, B_2 . Khi đó toán tử trùng – trật được xác định bởi một trong hai cách sau:

$$\text{Phép trùng - trật} - \text{HitMiss}(A, B_1, B_2) = \begin{cases} E(A, B_1) \cap E(A^c, B_2) \\ E(A, B_1) \bullet E(\bar{A}, B_2) \end{cases}$$

(161)

trong đó B_1 và B_2 đều bị chặn và tách biệt nhau (cần chú ý đến việc sử dụng ký hiệu lấy từ phương trình (81)). Hai tập hợp gọi là *tách biệt nếu* $B_1 \cap B_2 = \emptyset$. Xét theo một nghĩa

quan trọng thì *toán tử trùng - trật* là dạng tương đương về mặt morphology của kỹ thuật đối *sánh mẫu*, một kỹ thuật nổi tiếng dùng để đối sánh các khuôn dạng dựa trên mức độ tương quan chéo giữa chúng. Ở đây chúng ta có một mẫu B_1 cho đối tượng và một mẫu B_2 cho nền.

6.12.6. Tóm tắt những phép toán cơ bản

Kết quả ứng dụng của những phép toán cơ bản này được minh họa trong hình 41. Giá trị “-” đại diện cho ý “không quan tâm”. Cả ba phần tử cấu trúc này đều có tính đối xứng.

$$\begin{array}{ccc}
 B = N_8 = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} & B_1 = \begin{bmatrix} - & - & - \\ - & 1 & - \\ - & - & - \end{bmatrix} & B_2 = \begin{bmatrix} - & 1 & - \\ 1 & - & 1 \\ - & 1 & - \end{bmatrix} \\
 \text{(a)} & \text{(b)} & \text{(c)}
 \end{array}$$

Hình 41: Các phần tử cấu trúc B , B_1 , B_2 là các ma trận đối xứng 3×3

Kết quả xử lý được thể hiện trên hình 42, trong đó giá trị nhị phân “1” có màu đen, giá trị nhị phân “0” có màu trắng.

Phép toán mở có thể tách các đối tượng liên thông trong một ảnh nhị phân. *Phép toán đóng* có thể lấp đầy các lỗ nhỏ. Với một phần tử cấu trúc “trơn”, cả hai phép toán này đều cho một lượng trơn đáng kể trên biên của đối tượng nhận tác động từ chúng. *Phép mở* làm trơn biên đối tượng từ phía trong, còn *phép đóng* làm trơn biên đối tượng từ phía ngoài. Trong ví dụ về *phép trùng - trật*, phép toán này đã tìm ra được các điểm ảnh biên liên thông. Tuy nhiên, chúng ta còn có một phương pháp đơn giản khác để tìm biên của một đối tượng thông qua quan hệ sau:

$$\text{Biên liên thông 4} - \partial A = A - E(A, N_s) \quad (162)$$

hoặc:

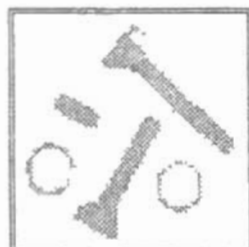
$$\text{Biên liên thông 8} - \partial A = A - E(A, N_d) \quad (163)$$



(a) Ảnh A



(b) Phép phát triển với 2B



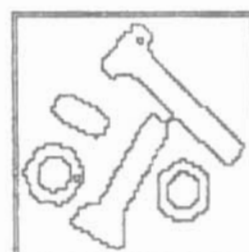
(c) Phép bào mòn với 2B



(d) Phép mở với 2B



(e) Phép đóng với 2B



(f) Phép trùng - trật với B_1 và B_2

Hình 42: Ví dụ về các phép toán morphology

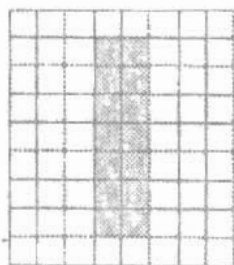
6.12.7 Phép rút xương

Định nghĩa không hình thức về xương của một đối tượng là: xương là một biểu diễn dạng đường của một đối tượng, trong đó:

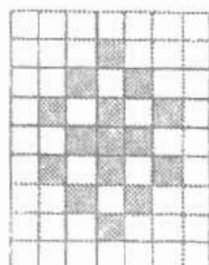
- i) Đường này có độ rộng một điểm ảnh,

- ii) Đường này đi qua phần “giữa” của đối tượng đó, và,
- iii) Đường này bảo toàn tô pô của đối tượng.

Những điều kiện nói trên không phải lúc nào cũng thực hiện được. Chúng ta có thể chỉ ra một tình huống như vậy trong hình 43.



(a)



(b)

Hình 43. Các phần ví dụ cho ba điều kiện ứng với khái niệm xương của đối tượng

Trong ví dụ đầu tiên ở hình 43 (a), để sinh ra một đường thể hiện sự đơn giản của đối tượng, chúng ta không thể nào tạo ra một đường thẳng có độ dày chỉ một điểm ảnh và nằm ở tâm của đối tượng. Trong hình 43(b), chúng ta lại không thể loại bỏ bất kỳ điểm ảnh nào của đối tượng liên thông 8 có trong hình và đồng thời bảo toàn tô pô của đối tượng, với đặc trưng tô pô ở đây chính là tính liên thông. Mặc dù vậy, chúng ta cũng có rất nhiều kỹ thuật cố gắng tạo ra được xương của đối tượng.

Công thức cơ bản cho việc tạo xương này dựa trên công trình của Lantúejoul [33]. Tập con xương $S_k(A)$ được xác định bởi:

Tập con xương -

$$S_k(A) = \{ \dots \} \quad k=0, 1, \dots, K \quad (165)$$

trong đó K là giá trị k lớn nhất trước khi $S_k(A)$ trở thành tập rỗng. Từ phương trình (156), chúng ta có $(E(A, kB) \circ B) \subseteq E(A, kB)$. Phần tử cấu trúc B được chọn trong Z^2 để xấp xỉ một đĩa tròn có tính lỗi, bị chặn và đối xứng. Khi đó *xương* sẽ là hội của các tập con *xương*:

$$\text{Xương} - S(A) = \bigcup_{k=0}^K S_k(A) \quad (166)$$

Dựa trên công thức này, chúng ta có thể thấy rằng đối tượng ban đầu là hoàn toàn có thể được tái xây dựng lại nếu những tri thức về các tập con $S_k(A)$, phần tử cấu trúc B , và giá trị K được cho trước:

$$\text{Tái xây dựng} - A = \bigcup_{k=0}^K (S_k(A) \oplus kB) \quad (167)$$

Mặc dù vậy, công thức vừa trình bày ở trên không bảo toàn được tô pô của đối tượng, là một trong những yêu cầu của quá trình tạo *xương*, như đã được nêu ra trong phương trình (164).

Một quan điểm khác trong tiếp cận giải quyết bài toán này là *phép làm mảnh*, hay chính xác là một phép bào mòn cho phép làm giảm độ dày của đối tượng nhưng không làm triệt tiêu nó. Thuật toán làm mảnh tổng quát có dạng sau:

$$\text{Phép làm mảnh} - \text{Thin}(A, B_1, B_2) = A - \text{HitMiss}(A, B_1, B_2) \quad (168)$$

Tùy theo cách chọn lựa B_1 và B_2 mà chúng ta sẽ có rất nhiều thuật toán làm mảnh khác nhau có thể được cài đặt.

Chúng ta còn có thể mô tả một cài đặt hết sức thực tế theo một cách khác. Nếu chúng ta tự giới hạn mình với lân

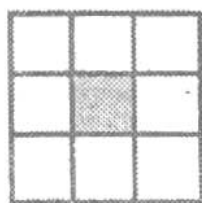
cận 3×3 , là lân cận tương tự với phần tử cấu trúc $B = N_8$ trong hình 41(a), khi đó phép toán làm mảnh có thể được xem như hành động quét đi quét lại một cửa sổ trên ảnh nhị phân. Điểm ảnh ở giữa cửa sổ này sẽ được nhận giá trị "0" dựa trên một số điều kiện nào đó. Điểm ảnh trung tâm này sẽ không bị đổi thành "0" nếu và chỉ nếu:

i) Có một điểm ảnh cô lập được tìm thấy (ví dụ như trong hình 44(a)).

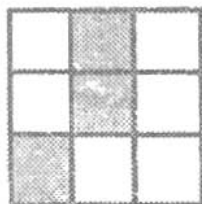
ii) Sự loại bỏ một điểm ảnh sẽ làm thay đổi tính liên thông (ví dụ như trong hình 44(b)).

iii) Sự loại bỏ một điểm ảnh sẽ làm thu ngắn một đường thẳng (ví dụ như trong hình 44 (c)).

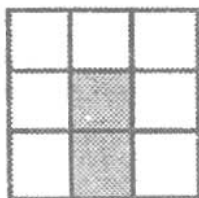
Quá trình loại bớt các điểm ảnh sau mỗi lần lặp được gọi là *phép bào mòn có điều kiện*. Ba trường hợp kiểm tra ứng với phương trình (169) được minh họa trong hình 44. Một cách tổng quát tất cả các phép quay và biến thể của chúng đều phải được kiểm tra. Do chỉ có tối đa 512 tổ hợp điểm ảnh ứng với cửa sổ 3×3 trên một ảnh nhị phân, chúng ta có thể cài đặt phép kiểm tra này dễ dàng bằng cách sử dụng bảng tra.



(a) Điểm ảnh cô lập



(b) Điểm ảnh liên thông



(c) Điểm ảnh cuối

Hình 44. Các điều kiện kiểm tra của phép bào mòn có điều kiện tại điểm ảnh trung tâm.

Nếu chỉ có điều kiện (i) được sử dụng thì mọi đối tượng đều sẽ bị bào mòn đến chỉ còn một điểm. Đây là cách chúng ta có thể sử dụng để đếm số lượng các đối tượng có trong ảnh. Nếu chỉ có điều kiện (ii) thì các lỗ trong các đối tượng sẽ được tìm thấy. Nếu điều kiện (i + ii) được sử dụng thì mỗi đối tượng đều sẽ bị bào mòn đến chỉ còn một điểm nếu như nó không chứa lỗ bên trong, hoặc sẽ bị bào mòn thành các vòng đóng nếu nó có chứa lỗ bên trong. Trong trường hợp cả ba điều kiện (i + ii + iii) được dùng thì một “khung xương hoàn chỉnh” sẽ được tạo ra dưới dạng một xấp xỉ của phương trình (56). Minh họa cho những tình huống này được thể hiện trong các hình 45(a) và 45(b).

6.12.8. Phép lan truyền

Trong thực tế, sẽ rất tiện lợi nếu như chúng ta có thể tái xây dựng lại một ảnh đã “sống còn” sau nhiều lần bị bào mòn, hoặc có thể làm đầy một đối tượng được xác định bởi đường biên của nó. Quá trình này có rất nhiều tên gọi như *lấp đầy vùng*, *tái xây dựng*, và *lan truyền*. Định nghĩa hình thức của nó được cho trong thuật toán sau đây. Chúng ta sẽ bắt đầu với một ảnh hạt giống $S^{(0)}$, một ảnh mặt nạ A , và một phần tử cấu trúc B . Sau đó chúng ta sẽ thực hiện nhiều phép phát triển S với B và che nó bởi mặt nạ A trong một thủ tục lặp như sau:

$$\text{Bước lặp } k - S^{(k)} = [S^{(k-1)} \oplus B] \cap A \text{ cho đến khi } S^{(k)} = S^{(k-1)}$$

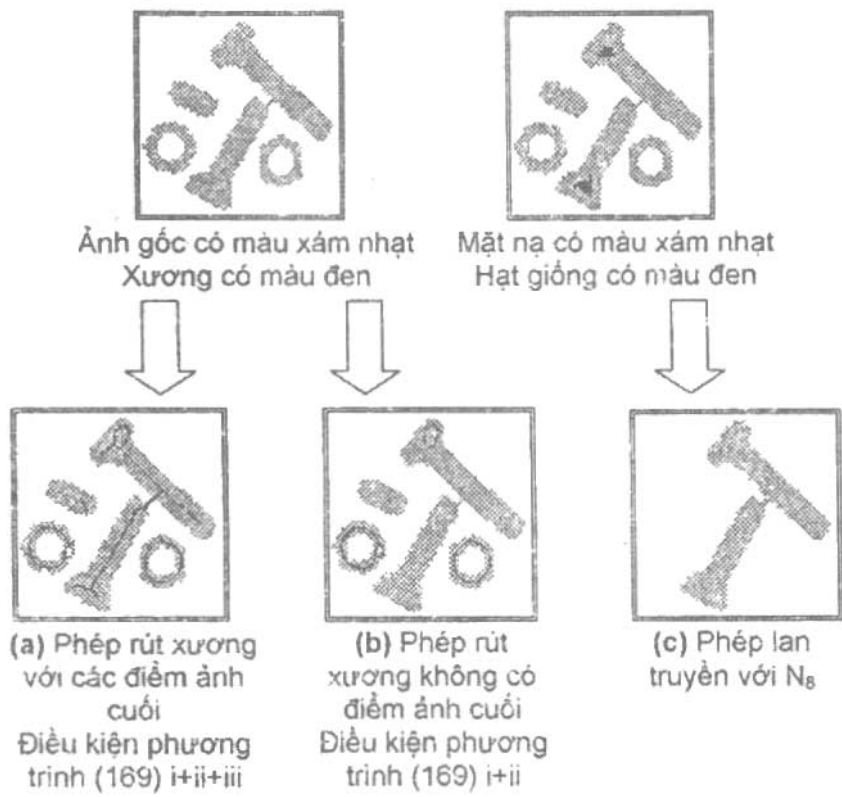
(170)

Sau mỗi lần lặp, ảnh hạt giống sẽ to thêm (do phép phát triển) nhưng vẫn nằm trong tập đối tượng xác định bởi A , hay nói cách khác, S lan truyền để lấp đầy A . Lựa

chọn thông dụng nhất đối với B và N_4 hay N_8 . Việc sử dụng phép lan truyền dẫn đến một số lưu ý sau đây. Trước hết, trong cách cài đặt trực tiếp theo định nghĩa, như đã được đề nghị trong phương trình (170), phí tổn tính toán là cực lớn. Mỗi bước lặp cần $O(N^2)$ phép toán cho một ảnh có kích thước $N \times N$, do vậy với một số bước lặp cần thiết độ phức tạp tính toán có thể đạt đến $O(N^3)$. May mắn thay là chúng ta còn có một cách cài đặt đệ quy của thuật toán này, trong đó chỉ cần quét qua toàn bộ ảnh một hoặc hai lần là đủ, tức là độ phức tạp tính toán chỉ tối đa là $O(N^2)$. Thứ hai là, mặc dù cho đến nay chúng ta vẫn chưa chú ý nhiều đến tính liên thông của đối tượng/ nền (xem hình, 37), rõ ràng tính liên thông được quy định bởi B là đối sánh được tính liên thông tương ứng trong định nghĩa về biên của A (xem các phương trình (162) và (163)). Cuối cùng, như đã được đề cập trước đây, việc lựa chọn chính xác các giá trị ("0" hay "1") là hết sức quan trọng cho điều kiện biên của ảnh. Sự lựa chọn này phụ thuộc vào ứng dụng.

6.12.9. Tóm tắt về phép rút xương & phép lan truyền

Ứng dụng của hai phép toán này trên ảnh thử nghiệm được minh họa trong hình 45.



Hình 45. Các ví dụ về phép rút xương và phép lan truyền

Trong hình 45(a) và 45(b) phép rút xương được thể hiện với hai tình huống: có điều kiện điểm ảnh cuối (phương trình (169) $i + ii + iii$) và không có điều kiện điểm ảnh cuối (phương trình (169) $i + ii$). Phép lan truyền được minh họa trong hình 45(c). Ảnh ban đầu, được thể hiện bằng màu xám nhạt, bị bào mòn bởi phép toán $E(A, 6N_8)$ để sinh ra ảnh hạt giống có màu đen. Ảnh ban đầu sau đó được sử dụng như một mặt nạ để cho ra kết quả sau cùng. Giá trị biên trong cả hai ảnh là "0".

Nhiều kỹ thuật đã được xây dựng dựa trên sự kết hợp *phép rút xương* và *phép lan truyền* cùng với nhiều phép toán morphology khác. Những kỹ thuật này sẽ được trình bày trong mục 7.3.3.

6.12.10. Xử lý giá trị xám theo kiểu morphology

Các kỹ thuật lọc morphology có thể mở rộng cho các ảnh xám. Để đơn giản chúng ta sẽ giới hạn phần trình bày ở đây trên các phần tử cấu trúc B chứa một số hữu hạn điểm ảnh, có tính lỗi và bị chặn. Các phần tử cấu trúc này có các giá trị xám liên kết với mỗi vị trí tọa độ như trong ảnh A .

Phép phát triển mức xám

Ký hiệu là $D_G(*)$ và được xác định bởi:

$$D_G(A, B) = \max_{\{j,k\} \in B} \{a[m-j, n-k] + b[j, k]\} \quad (171)$$

Với một tọa độ xuất cho trước $[m, n]$, phần tử cấu trúc sẽ được lấy tổng với từng phiên bản dịch chuyển của ảnh, giá trị tối đa nhận được trên tất cả các phép dịch trong vùng $J \times K$ của B sẽ được dùng làm kết quả sau cùng. Trong trường hợp phép dịch chuyển cần đến những giá trị của ảnh A nằm ngoài vùng $M \times N$, sẽ có một quyết định được đưa ra để chọn mô hình ảnh mở rộng nên sử dụng, như đã đề cập đến trong mục 6.9.2.

Phép bào mòn mức xám

Ký hiệu là $E_G(*)$ và được xác định bởi:

$$E_G(A, B) = \min_{\{j,k\} \in B} \{a[m+j, n+k] - b[j, k]\} \quad (172)$$

Tính đối ngẫu giữa các phép toán *bào mòn mức xám* và *phát triển mức xám* phức tạp hơn so với trường hợp ảnh nhị phân:

$$\begin{aligned} E_G(A, B) &= -D_G(-A, \bar{B}) \\ D_G(A, B) &= -E_G(-A, \bar{B}) \end{aligned} \quad (173)$$

trong đó “ \bar{A} ” có nghĩa là $a[j, k] \rightarrow a[-j, -k]$, còn có “ $-A$ ” có nghĩa là $a[j, k] \rightarrow -a[j, k]$.

Sau đây là định nghĩa của hai phép toán bậc cao hơn, *phép mở mức xám* và *phép đóng mức xám*:

$$\text{Phép mở mức xám: } -O_G(A, B) = D_G(E_G(A, B), B) \quad (174)$$

$$\text{Phép đóng mức xám } -C_G(A, B) = E_G(D_G(A, \bar{B}), \bar{B}) \quad (175)$$

Tính đối ngẫu giữa *phép mở mức xám* và *phép đóng mức xám* là:

$$\begin{aligned} O_G(A, B) &= -C_G(-A, B) \\ C_G(A, B) &= -O_G(-A, B) \end{aligned} \quad (176)$$

Những tính chất quan trọng đã được trình bày trong các mục trước, như tính dừng, tính bất biến trong tịnh tiến, tính tăng trong $A...$ cũng áp dụng được cho việc xử lý các giá trị xám theo kiểu morphology. Độc giả quan tâm đến chi tiết về những chủ đề này có thể tham khảo trong sách của Giardina và Dougherty [6].

Trong nhiều tình huống, độ phức tạp tính toán của việc xử lý mức xám theo kiểu morphology được giảm đi đáng kể thông qua việc sử dụng những phần tử cấu trúc đối xứng trong đó $b[j, k] = b[-j, -k]$. Thông dụng nhất trong số những phần tử kiểu này là $B = \text{constant} = 0$. Ở đây bằng cách sử dụng lại vùng $[j, k] \in B$, các định nghĩa trên sẽ được rút gọn thành:

Phép phát triển mức xám -

$$D_G(A, B) = \max_{\{j,k\} \in B} \{a[m-j, n-k]\} = \max_B(A) \quad (177)$$

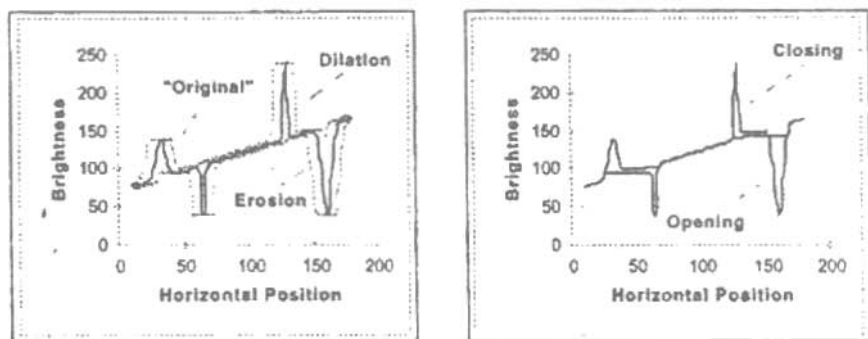
Phép bào mòn mức xám

$$E_G(A, B) = \min_{\{j,k\} \in B} \{a[m-j, n-k]\} = \min_B(A) \quad (178)$$

Phép mở mức xám - $O_G(A, B) \max(\min(A))$ (179)

Phép đóng mức xám - $C_G(A, B) \min(\max(A))$ (180)

Chúng ta có một kết luận đáng quan tâm là các bộ lọc cực đại và bộ lọc cực tiểu, như đã giới thiệu trong mục 6.10.2, chính là các bộ lọc phát triển và bào mòn mức xám với một phần tử cấu trúc xác định nhờ hình dáng của cửa sổ lọc, trong đó giá trị xám "0" ở bên trong cửa sổ. Ví dụ về những phép toán này trên tín hiệu đơn giản 1-chiều được minh họa trong hình 46.



Hình 46. *Lọc morphology của dữ liệu mức xám*

Với một cửa sổ chữ nhật $J \times K$, một bộ lọc cực đại hoặc cực tiểu hai chiều có thể tách được thành hai cửa sổ một

chiều. Hơn nữa, một bộ lọc cực đại hoặc cực tiểu một chiều có thể được viết lại dưới dạng từng bước (xem mục 6.9.2). Điều này có nghĩa là các phép phát triển và bào mòn mức xám có độ phức tạp tính toán trên mỗi điểm ảnh là $O(\text{constant})$, tức là độc lập với J và K (xem thêm bảng 13).

Các phép toán được định nghĩa trên đây có thể dùng để tạo nên nhiều thuật toán morphology giúp làm trơn ảnh, xác định gradient và toán tử Laplace. Tất cả chúng đều được xây dựng từ hai thành tố cơ bản là *phép phát triển* mức xám và *phép bào mòn* mức xám. Trong mọi trường hợp các bộ lọc cực đại và cực tiểu đều lấy giá trị trong vùng $[j,k] \in B$.

6.12.11. Làm trơn theo kiểu morphology

Thuật toán này dựa trên sự quan sát rằng phép mở mức xám làm trơn một ảnh mức xám từ phía trên của mặt độ sáng được cho trước bởi hàm $a[m,n]$, còn *phép đóng mức xám* làm trơn từ phía dưới. Chúng ta sử dụng một phần tử cấu trúc B dựa trên các phương trình (84) và (178).

$$\begin{aligned} \text{MorphSmooth}(A,B) &= C_G(O_G(A,B),B) \\ &= \min(\max(\max(\min(A)))) \quad (181) \end{aligned}$$

Chú ý rằng ở công thức trên chúng ta đã thay thế ký hiệu phần tử cấu trúc B bởi các phép toán \max và \min nhằm giúp cho việc biểu diễn ký hiệu đơn giản hơn.

6.12.12. Tính gradient theo kiểu morphology

Trong số các bộ lọc tuyến tính, bộ lọc gradient có biểu diễn dạng vector (phương trình (103)) với một giá trị độ lớn (phương trình (104)) và một hướng *phương trình (105)). Phiên bản mà chúng ta trình bày ở đây sinh ra một ước lượng kiểu morphology cho độ lớn của gradient:

$$\begin{aligned}
 \textit{Gradient}(A, B) &= \frac{1}{2} (D_G(A, B) - E_G(A, B)) \\
 &= \frac{1}{2} (\max(A) - \min(A)) \quad (182)
 \end{aligned}$$

6.12.13. Toán tử Laplace theo kiểu morphology

Bộ lọc Laplace theo kiểu morphology được cho bởi:

$$\begin{aligned}
 \textit{Laplacian}(A, B) &= \frac{1}{2} ((D_G(A, B) - A - (A - E_G(A, B))) \\
 &= \frac{1}{2} (D_G(A, B) + E_G(A, B) - 2A) \quad (183) \\
 &= \frac{1}{2} (\max(A) - \min(A) - 2A)
 \end{aligned}$$

6.12.14. Tóm tắt về các bộ lọc morphology

Tác động của những bộ lọc này được minh họa trong hình 46. Tất cả các ảnh đều được xử lý với phần tử cấu trúc 3×3 như đã mô tả trong các phương trình từ (177) đến (183). Hình 47(e) đã được co giãn về độ tương phản nhờ phương trình (78) với các tham số 1% và 99%. Hình 47(c), 47(d), 47(e) nên được so sánh với các hình 31, 33 và 34 khi chúng ta quan sát chúng.



(a) Phép phát triển



(b) Phép bào mòn



(c) Làm tròn



(d) Gradient



(e) Laplace

Hình 47. Ví dụ về các bộ lọc morphology mức xám

Chương 7

MỘT SỐ KỸ THUẬT HIỆU CHỈNH & PHÂN ĐOẠN ẢNH

Các thuật toán đã giới thiệu trong chương 6 có thể được dùng để xây dựng nên nhiều kỹ thuật giải quyết các bài toán xử lý ảnh cụ thể. Trong chương này không trình bày hết những kỹ thuật ấy, mà chỉ tập trung vào ba nhóm kỹ thuật quan trọng nhất là:

- Kỹ thuật hiệu chỉnh tạo báo bóng;
- Các kỹ thuật phục hồi và tăng cường ảnh cơ bản;
- Kỹ thuật phân đoạn.

7.1. KỸ THUẬT HIỆU CHỈNH SỰ TẠO BÓNG

Phương pháp sinh ra các ảnh số, trong đó có sự tương tác giữa các đối tượng trong không gian thực tế, sự chiếu sáng, và camera, thường dẫn đến tình huống ảnh có nhiều độ bóng trải khắp và vùng nhìn. Trong một số trường hợp, ánh sáng lên ở tâm và mờ dần về phía các cạnh. Trong một số trường hợp khác, ánh trở nên tối hơn ở vùng bên trái và sáng hơn ở vùng bên phải. Những độ bóng như vậy có thể là do sự rọi sáng không đều, do độ nhạy không đều của camera, hay ngay cả do các vết bẩn và bụi bám trên bề mặt kính hay thấu kính của những thiết bị tạo ảnh. Nói chung, hiệu ứng tạo bóng này thường không nằm trong sự mong muốn của người tạo ra ảnh. Do vậy, việc triệt tiêu nó thường là cần thiết cho

những bước xử lý tiếp sau, đặc biệt khi mục đích cuối cùng của quá trình xử lý ảnh là nhằm để phân tích hay hiểu ảnh.

7.1.1. Mô hình tạo bóng

Một cách tổng quát, chúng ta sẽ bắt đầu với sự giới thiệu về mô hình tạo bóng. Trong mô hình này, độ rọi sáng $I_{III}(x,y)$ thường tương tác với đối tượng $a(x,y)$ để cho ra ảnh $b(x,y)$ theo quan hệ sau đây:

$$b(x,y) = I_{III}(x,y) \bullet a(x,y) \quad (184)$$

với đối tượng thể hiện nhiều mô hình ảnh như sau:

$$a(x,y) = \begin{cases} r(x,y) & \text{mô hình phản chiếu} \\ 10^{-OD(x,y)} & \text{mô hình thẩm thấu} \\ c(x,y) & \text{mô hình huỳnh quang} \end{cases} \quad (185)$$

trong đó tại vị trí (x,y) , $r(x,y)$ là độ phản chiếu, $OD(x,y)$ là độ *trụ mật quang học*, và $c(x,y)$ là độ tập trung của vật liệu huỳnh quang. Chúng ta chú ý rằng mô hình huỳnh quang chỉ đúng với độ tập trung huỳnh quang thấp. Camera sau đó có thể góp thêm hai thuật ngữ mới là *gain* và *offset*, như trong phương trình (74), sao cho:

$$\begin{aligned} & \text{Tạo bóng toàn thể -} \\ c[m,n] &= gain[m,n] \bullet b[m,n] + offset[m,n] \\ &= gain[m,n] \bullet I_{III}[m,n] \bullet a[m,n] + offset[m,n] \end{aligned} \quad (186)$$

Chúng ta giả sử rằng $I_{III}[m,n]$ biến đổi chậm hơn so với $a[m,n]$.

7.1.2 Ước lượng sự tạo bóng

Chúng ta cần phân biệt hai trường hợp xác định $a[m,n]$ bắt đầu từ $c[m,n]$. Trong cả hai trường hợp, chúng ta đều định

ước lượng các thuật ngữ tạo bóng $\{gain[m,n] \cdot I_{III}[m,n]\}$ và $\{offset[m,n]\}$. Trong trường hợp đầu, chúng ta giả sử chỉ có được ảnh đã ghi $c[m,n]$ để xử lý, còn trong trường hợp thứ hai chúng ta giả sử rằng có thể ghi thêm hai ảnh kiểm tra.

*** Ước lượng sau**

Trong trường hợp này chúng ta sẽ cố gắng rút trích ước lượng tạo bóng từ $c[m,n]$. Các khả năng phổ biến nhất được trình bày dưới đây.

Lọc thông thấp

Chúng ta đi tính một phiên bản đã làm trơn của $c[m,n]$ tại vị trí độ trơn là khá lớn so với kích thước của các đối tượng trong ảnh. Phiên bản này được đưa ra với ý định lấy đó làm một ước lượng cho nền ảnh. Sau đó, nó sẽ bị loại khỏi $c[m,n]$ và giá trị DC mong muốn sẽ được phục hồi. Quá trình này được thể hiện qua công thức sau:

$$\hat{a}[m,n] = c[m,n] - \text{LowPass} \{c[m,n]\} + \text{const} \quad (187)$$

trong đó $\hat{a}[m,n]$ là ước lượng của $a[m,n]$. Ở đây, việc chọn được bộ lọc thông thấp thích hợp đồng nghĩa với việc biết được các tần số tọa độ phù hợp trong miền Fourier nơi các thuật ngữ tạo bóng chiếm ưu thế.

Lọc đồng hình

Chúng ta cần lưu ý rằng, nếu $offset[m,n] = 0$, thì $c[m,n]$ chỉ bao gồm các số lượng được nhân lên nhiều lần. Hơn nữa, số hạng $\{gain[m,n] \cdot I_{III}[m,n]\}$ thay đổi chậm trong khi $a[m,n]$ được giả sử trước là không thay đổi chậm như vậy. Do vậy chúng ta lấy logarit của $c[m,n]$ để tạo ra hai số hạng, một có tần số thấp và một có tần số cao.

Chúng ta loại bỏ sự tạo bóng bằng cách lọc thông cao logarit của $c[m,n]$, rồi lại lấy lũy thừa (lấy nghịch đảo logarit) để phục hồi lại ảnh. Thuật toán này dựa trên kỹ thuật *lọc đồng hình* do Oppenheim, Schafer và Stockham phát triển trong [34]. Chúng ta có biểu diễn dưới dạng công thức như sau:

$$i) c[m,n] = \text{gain}[m,n] \bullet I_{III}[m,n] \bullet a[m,n]$$

$$ii) \ln \{c[m,n]\} = \ln \left\{ \underbrace{\text{gain}[m,n] \bullet I_{III}[m,n]}_{\text{Biến đổi chậm}} \right\} + \left\{ \underbrace{a[m,n]}_{\text{Biến đổi nhanh}} \right\}$$

$$iii) \text{HighPass} \{ \ln \{c[m,n]\} \} \approx \ln \{a[m,n]\} \quad (188)$$

$$iv) a[m,n] = \exp \{ \text{HighPass} \{ \ln \{c[m,n]\} \} \}$$

Lọc morphology

Một lần nữa chúng ta lại đi tính một phiên bản đã làm trơn của $c[m,n]$ tại nơi có độ trơn khá lớn so với kích thước các đối tượng trong ảnh, nhưng lần này kỹ thuật làm trơn theo kiểu morphology sẽ được sử dụng, như đã trình bày trong phương trình (187). Phiên bản đã làm trơn này là một ước lượng cho nền ảnh. Sau đó, chúng ta sẽ loại nó ra khỏi $c[m,n]$ và phục hồi giá trị DC mong muốn. Quá trình này được thể hiện qua công thức sau:

$$\hat{a}[m,n] = c[m,n] - \text{MorphSmooth} \{c[m,m]\} + \text{const} \quad (189)$$

Ở đây việc chọn được cửa sổ lọc morphology thích hợp đồng nghĩa với việc biết được (hoặc ước lượng được) kích thước của những đối tượng quan tâm lớn nhất trong ảnh.

Ước lượng ưu tiên

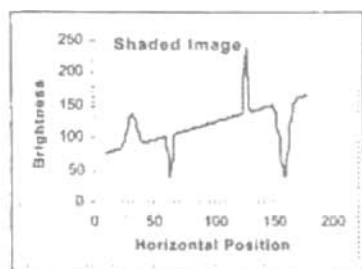
Nếu các ảnh kiểm định có thể được ghi nhận lại thông qua hệ thống camera, thì kỹ thuật thích hợp nhất thể loại bỏ tác động tạo bóng là ghi lại hai ảnh – $BLACK[m,n]$ và $WHITE[m,n]$. Ảnh $BLACK$ được tạo ra bằng cách che các thấu kính dẫn đến $b[m,n] = 0$, qua đó cho $BLACK[m,n] = offset[m,n]$. Ảnh $WHITE$ được sinh ra bằng cách sử dụng $a[m,n] = 1$, khi đó chúng ta có $WHITE[m,n] = gain[m,n] \cdot I_{III}[m,n] + offset[m,n]$. Kỹ thuật hiệu chỉnh sẽ được cho bởi phương trình sau:

$$d[m,n] = constant \cdot \frac{c[m,n] - BLACK[m,n]}{WHITE[m,n] - BLACK[m,n]} \quad (190)$$

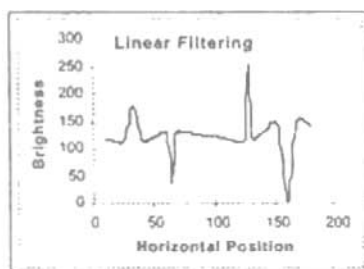
trong đó số hạng constant được chọn để sinh ra một thang độ mong muốn.

Tác động của những kỹ thuật nói trên đối với dữ liệu của hình 46 được chỉ ra trong hình 48. Sự tạo bóng ở đây có dạng một đoạn dốc tuyến tính tăng từ trái sang phải, còn các đối tượng chứa những đỉnh Gauss lại có bề rộng thay đổi.

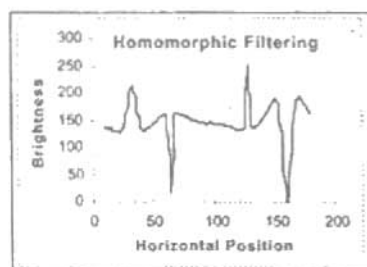
Một cách tóm tắt, nếu có thể thu được các ảnh kiểm định $BLACK$ và $WHITE$, thì phương trình (190) sẽ được sử dụng. Trong trường hợp không thu được các ảnh như vậy, chúng ta sẽ sử dụng một trong số những thuật toán còn lại.



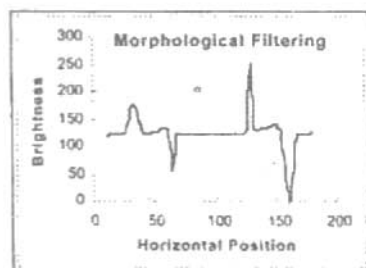
(a) Ảnh ban đầu



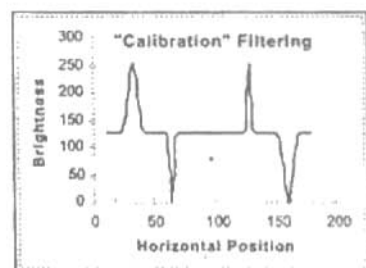
(b) Hiệu chỉnh với lọc thông thấp



(c) Hiệu chỉnh với lọc logarit



(c) Hiệu chỉnh với lọc Max/Min



(d) Hiệu chỉnh với ảnh kiểm tra

Hình 48. So sánh các thuật toán hiệu chỉnh sự tạo bóng.
 Kết quả cuối cùng (d) đồng nhất với ảnh ban đầu.

7.2. CÁC KỸ THUẬT PHỤC HỒI VÀ TĂNG CƯỜNG ẢNH CƠ BẢN

Quá trình thu nhận ảnh thường dẫn đến sự thoái hóa chất lượng của ảnh. Do nhiều nguyên nhân khác nhau như một số vấn đề về cơ học, chuyển động, sự rọi sáng không phù hợp, và nhiễu, mà chất lượng của ảnh số hóa có thể giảm đi so với ảnh ban đầu. Mục đích của việc tăng cường ảnh là – bắt đầu từ một ảnh đã được ghi nhận $c[m,n]$ – cho ra ảnh mới tốt nhất về mặt trực quan $\hat{a}[m,n]$. Trong khi đó, phục hồi ảnh – cũng bắt đầu từ một ảnh đã được ghi nhận $c[m,n]$ – lại cho ra ước lượng tốt nhất có thể có $\hat{a}[m,n]$ của ảnh ban đầu $a[m,n]$. Như vậy tăng cường ảnh quan tâm đến vẻ đẹp, còn phục hồi ảnh quan tâm đến tính xác thực của dữ liệu ảnh.

Để đánh giá mức độ thành công trong việc phục hồi ảnh, người ta thường sử dụng một độ đo lỗi giữa ảnh ban đầu $a[m,n]$ và ảnh ước lượng $\hat{a}[m,n]$, ký hiệu là $E\{\hat{a}[m,n], a[m,n]\}$. Cho đến nay vẫn không ai biết được là hàm sai số toán học nào lại có thể biểu diễn được sự nhận thức của con người về mức độ khác biệt giữa hai ảnh. Người ta vẫn thường sử dụng hàm sai số bình phương trung bình vì những lý do sau đây:

- Nó dễ tính toán;
- Nó khả vi, do vậy có thể tìm được cực tiểu;
- Nó tương ứng với “năng lượng tín hiệu” trong sai số tổng thể;
- Nó có những tính chất hay liên quan đến định lý Parseval đã được đề cập đến trong các phương trình (22) và (23).

Sai số bình phương trung bình được xác định bởi:

$$E\{\hat{a}, a\} = \frac{1}{MN} \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} \{\hat{a}[m,n] - a[m,n]\}^2 \quad (191)$$

Trong một số kỹ thuật phục hồi và tăng cường ảnh thì độ đo lỗi là không cần thiết, nhưng trong một số kỹ thuật khác nó lại cần phải có để đánh giá và so sánh những kết quả đạt được.

7.2.1 Kỹ thuật tạo mặt nạ không sắc nét

Chúng ta có một kỹ thuật nổi tiếng lấy từ kỹ thuật chụp hình nhằm cải thiện chất lượng trực quan của một ảnh. Đó là kỹ thuật nâng cao các cạnh của ảnh, được biết đến với tên gọi *kỹ thuật tạo mặt nạ không sắc nét*. Nâng cao cạnh có nghĩa là trước hết các cạnh trong ảnh sẽ bị cô lập, khuếch đại, và sau đó thêm ngược trở lại vào ảnh. Bằng cách quan sát cẩn thận hình 34, chúng ta thấy rằng toán tử Laplace là một cơ chế cho phép cô lập các cạnh trong ảnh sẽ bị cô lập, khuếch đại, và sau đó thêm ngược trở lại vào ảnh. Bằng cách quan sát cẩn thận hình 34, chúng ta thấy rằng toán tử Laplace là một cơ chế cho phép cô lập các cạnh mức xám. Điều này ngay lập tức dẫn tới kỹ thuật mà chúng ta đang cần:

$$\hat{a}[m,n] = a[m,n] + (k \bullet \nabla^2 a[m,n]) \quad (192)$$

Ảnh ban đầu



Ảnh sau khi đã
nâng cao nhờ
toán tử Laplace

Hình 49. So sánh giữa ảnh gốc và ảnh có cạnh được nâng cao

Số hạng k ở đây gọi là số hạng khuếch đại ($k > 0$).

Kết quả của kỹ thuật này được chỉ ra trong hình 49. Trong hình này, toán tử Laplace được dùng để tạo ra ảnh theo phương trình (120) với hệ số khuếch đại $k = 1$.

7.2.2. Khử nhiễu

Các kỹ thuật khử nhiễu hiện có được chia ra làm hai nhóm, một nhóm dựa trên thông tin về thời gian, còn nhóm còn lại dựa trên thông tin về tọa độ. Dựa trên thông tin thời gian có nghĩa là chúng ta có một dãy ảnh $\{a_p[m, n] \mid p = 1, 2, \dots, P\}$ có sẵn và chứa chính xác các đối tượng như nhau. Những đối tượng này chỉ khác nhau về mặt hiện thực hóa nhiễu độc lập. Khi đó nếu nhiễu có tính chất cộng tính, thì giá trị trung bình của dãy được cho bởi:

$$\bar{a}[m, n] = \frac{1}{P} \sum_{p=1}^P a_p[m, n] \quad (193)$$

sẽ tạo ra một kết quả trong đó giá trị trung bình của mỗi điểm ảnh không thay đổi. Với mỗi điểm ảnh, độ lệch chuẩn sẽ giảm từ σ đến $\frac{\sigma}{\sqrt{P}}$.

Nếu việc lấy trung bình theo thời gian không thể có được, thì bằng cách lấy trung bình theo tọa độ chúng ta cũng có thể làm giảm nhiễu trong ảnh. Tuy nhiên phương pháp này thường phải trả giá bằng bộ sắc nét của ảnh. Bốn lựa chọn dễ thấy nhất cho việc lấy trung bình theo tọa độ là các thuật toán làm trơn như đã mô tả trong mục 6.10, bao gồm: lọc Gauss theo phương trình (93), lọc trung vị, lọc Kuwahara, và làm trơn theo kiểu morphology như đã trình bày trong phương trình (181).

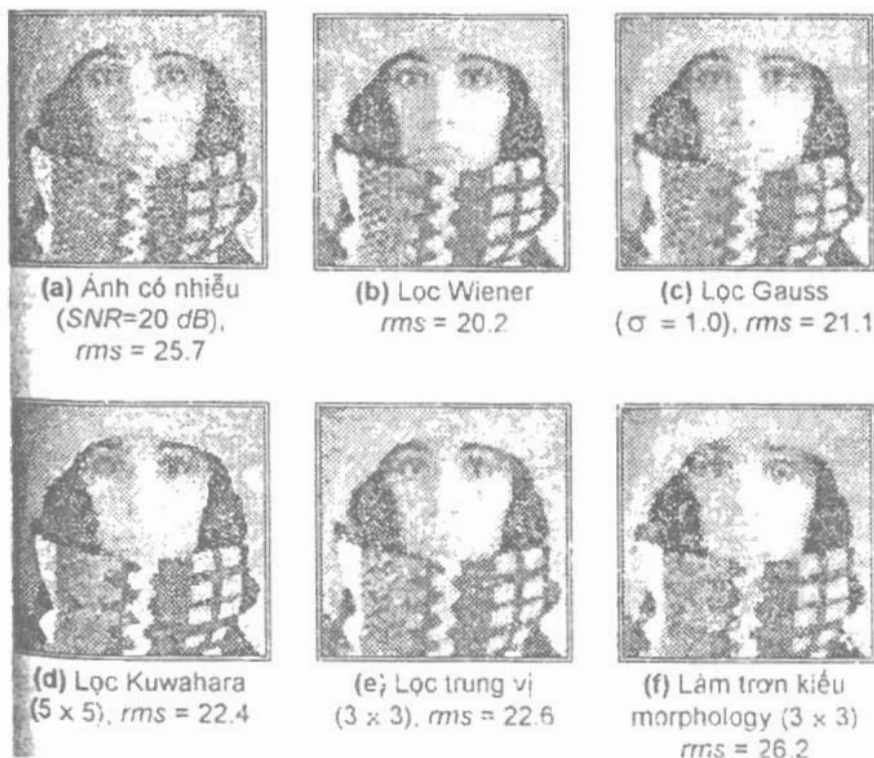
Trong số các bộ lọc tuyến tính, bộ lọc tối ưu trong phục hồi ảnh có nhiều là bộ lọc Wiener [2]. Từ “tối ưu” được dùng ở đây theo nghĩa bộ lọc này sẽ làm cực tiểu sai số bình phương trung bình (mse). Do phép toán lấy căn bậc hai có tính đơn điệu tăng, nên bộ lọc tối ưu này cũng làm cực tiểu căn bậc hai của sai số bình phương trung bình (rms). Bộ lọc Wiener được đặc trưng hóa trong miền Fourier và với nhiều cộng tính độc lập với tín hiệu, nó sẽ được xác định bằng phương trình sau đây:

$$H_W(u, v) = \frac{S_{aa}(u, v)}{S_{aa}(u, v) + S_{nn}(u, v)} \quad (194)$$

trong đó $S_{aa}(u, v)$ là độ trừ mật phổ năng lượng của các ảnh ngẫu nhiên $\{a[m, n]\}$, còn $S_{nn}(u, v)$ là độ trừ mật phổ năng lượng của nhiễu ngẫu nhiên. Nếu chúng ta có một ảnh đơn thì $S_{aa}(u, v) = |A(u, v)|^2$. Trong thực tế chúng ta rất khó có được độ trừ mật phổ năng lượng của ảnh chưa bị làm bẩn. Do nhiều ảnh có độ trừ mật phổ năng lượng tương tự nhau, và có thể được mô hình hóa bởi Bảng 11-T.8, nên mô hình đó có thể dùng để cung cấp một ước lượng cho $S_{aa}(u, v)$.

Hình 50 so sánh năm kỹ thuật khác nhau vừa được mô tả ở trên. Bộ lọc Wiener được xây dựng trực tiếp từ phương trình (113) do phổ ảnh và phổ nhiễu đều đã được biết. Các tham số cho những bộ lọc khác được chọn sao cho giá trị σ hay kích thước của số lọc đều dẫn đến sự cực tiểu của rms.

Căn bậc hai của sai số bình phương trung bình (rms) tương ứng với các bộ lọc khác nhau cũng được chỉ ra trong hình 50. Trong phép so sánh đặc biệt này, bộ lọc Wiener sinh ra sai số thấp hơn sai số do các bộ lọc khác tạo ra. Hai thủ



Hình 50. Khử nhiễu bằng các kỹ thuật lọc khác nhau

tục tuyến tính là lọc Wiener và lọc Gauss cho kết quả tương đối tốt hơn một chút so với ba bộ lọc phi tuyến còn lại.

7.2.3. Khử độ méo của ảnh

Mô hình đã được thể hiện ở trên đây – một ảnh bị làm sai lệch bởi nhiễu – nói chung, chưa đủ tinh tế để thể hiện được bản chất thật sự của sự sai lệch xuất hiện trong một ảnh số hóa. Một mô hình thực tế hơn cần có không những sự hiện diện của nhiễu mà còn có sự méo mó ảnh gây ra bởi các thấu kính, bởi độ mở của camera, bởi chuyển động của camera

và/hoặc một đối tượng, cùng nhiều thứ khác nữa. Người ta thường sử dụng mô hình trong đó ảnh $a[m,n]$ bị làm méo mó đi bởi một hệ thống dịch chuyển bất biến và tuyến tính $h_d[m,n]$ (chẳng hạn như các thấu kính) và bị làm bản đi bởi nhiễu $k[m,n]$. Trong các chương trước chúng ta đã thảo luận nhiều khía cạnh lý thuyết có liên quan đến $h_d[m,n]$ và $k[m,n]$. Sự kết hợp phổ biến nhất của các yếu tố vừa nêu chính là mô hình công tính cho bởi:

$$c[m,n] = (a[m,n] \otimes h_0[m,n]) + k[m,n] \quad (195)$$

Ở đây chủ tịch lọc hồi ảnh nhờ lọc tuyến tính, đi kèm với một tiêu chuẩn về số số bình phương trung bình một lần nữa lại cho chúng ta bộ lọc Wiener:

$$\begin{aligned} H_W(u, v) &= \frac{H_0(u, v) S_{aa}(u, v)}{|H_0(u, v)|^2 S_{aa}(u, v) + S_{nn}(u, v)} \\ &= \frac{H_0^*(u, v)}{|H_0(u, v)|^2 + \frac{S_{nn}(u, v)}{S_{aa}(u, v)}} \end{aligned} \quad (196)$$

Một lần nữa $S_{aa}(u, v)$ và độ trừ mật phổ năng lượng của ảnh, $S_{nn}(u, v)$ là độ trừ mật phổ năng lượng của nhiễu, còn $H_0(u, v) = F\{h_d[m, n]\}$.

Sự khảo sát công thức này trong một số trường hợp đặc biệt là hết sức hữu ích. Với những tần số tại đó $S_{aa}(u, v) \gg S_{nn}(u, v)$, tức là khi phổ tín hiệu chiếm ưu thế so với phổ nhiễu, bộ lọc Wiener được xác định bởi $\frac{1}{H_0(u, v)}$ là lời giải của bộ lọc ngược. Với những tần số tại đó $S_{aa}(u, v) \ll S_{nn}(u, v)$ tức là khi phổ nhiễu chiếm ưu thế so với phổ tín hiệu, bộ lọc

Wiener sẽ tỷ lệ với $H_0^*(u,v)$ là lời giải của bộ lọc đối sánh. Với những tần số tại đó $H_0(u,v) = 0$, bộ lọc Wiener $H_W(u,v) = 0$ sẽ ngăn chặn tình huống tràn.

Bộ lọc Wiener là một giải pháp cho bài toán phục hồi ảnh dựa trên sự sử dụng có tính giả thiết một bộ lọc tuyến tính và một tiêu chuẩn về sai số bình phương trung bình cực tiểu (hay rms). Trong ví dụ dưới đây, ảnh $a[m,n]$ bị làm méo bởi một bộ lọc thông dải và sau đó nhiễu trắng được thêm vào để tạo ra SNR – 30 dB. Hình 51 thể hiện các kết quả ứng dụng lọc Wiener và lọc trung vị so với ảnh bị nhiễu ban đầu.



(a) Ảnh méo có nhiễu



(b) Lọc Wiener
rms = 108.4



(c) Lọc trung vị
(3 × 3), rms = 40.9

Hình 51. Khử nhiễu và độ méo ảnh nhờ bộ lọc Wiener theo phương trình (196) và nhờ bộ lọc trung vị

Giá trị rms sau khi lọc Wiener và trước khi co giãn độ tương phản là 108.4. Sau khi co giãn độ tương phản với [(77) thì kết quả sau cùng trong hình 51(b) có sai số bình phương trung bình là 27.8. Bằng cách sử dụng một bộ lọc trung vị 3×3 như trong hình 51(c) sẽ dẫn đến rms có giá trị 40.9 trước khi co giãn, còn sau khi co giãn là 35.1. Mặc dù bộ lọc Wiener cho sai số rms cực tiểu trong số các bộ lọc tuyến tính, bộ lọc trung vị phi tuyến còn cho giá trị rms nhỏ hơn nữa.

Phép toán *co giãn độ tương phản* bản thân nó là một phép toán phi tuyến. “Chất lượng trực quan” của kết quả lọc trung vị hoàn toàn có thể sánh ngang với kết quả học Wiener. Đó là do những yếu tố nhân tạo có tính tuần hoàn sinh bởi một bộ lọc tuyến tính, mà chúng ta có thể quan sát thấy trên hình 51(b).

7.3. PHÂN ĐOẠN ẢNH

Để phân tích các đối tượng trong ảnh, chúng ta cần phải phân biệt được các đối tượng cần quan tâm với phần còn lại của ảnh, hay còn gọi là nền ảnh. Những đối tượng này có thể tìm ra được nhờ các *kỹ thuật phân đoạn ảnh*, theo nghĩa tách phần tiền cảnh ra khỏi hậu cảnh trong ảnh. Trong chương này, chúng ta sẽ tìm hiểu những kỹ thuật phổ biến nhất trong phân đoạn ảnh, đó là *kỹ thuật lấy ngưỡng* và *kỹ thuật dò cạnh*, cùng với những phương pháp giúp cải thiện chất lượng của quá trình phân đoạn. Chúng ta cần phải hiểu được là:

- Không có kỹ thuật phân đoạn nào là vạn năng, theo nghĩa có thể áp dụng cho mọi loại ảnh.
- Không có kỹ thuật phân đoạn nào là hoàn hảo.

7.3.1. Kỹ thuật lấy ngưỡng

Kỹ thuật này dựa trên một ý tưởng hết sức đơn giản. Một tham số θ , gọi là ngưỡng độ sáng, sẽ được chọn để áp dụng cho một ảnh $a[m,n]$, theo cách sau:

$$\begin{aligned} \text{If } a[m,n] \geq \theta & \quad a[m,n] = \text{object} = 1 \\ \text{Else} & \quad a[m,n] = \text{background} = 0 \end{aligned} \quad (197)$$

Thuật toán trên giả định rằng chúng ta đang quan tâm đến các đối tượng sáng (*object*) trên nền tối (*background*). Với

các đối tượng tối trên nền sáng chúng ta có thể dùng thuật toán sau:

$$\begin{aligned} \text{If } a[m,n] < \theta & \quad a[m,n] = \text{object} = 1 \\ \text{Else} & \quad a[m,n] = \text{background} = 0 \end{aligned} \quad (198)$$

Kết quả của các thuật toán trên đây là sự thay thế các nhãn “đối tượng” (object) hay “nền ảnh” (background) bằng các giá trị logic “1” hoặc “0”. Về nguyên tắc, điều kiện kiểm tra trong thuật toán nên dựa trên một đặc trưng quan trọng nào đó khác hơn là độ sáng, do đây là một đặc trưng khá đơn giản (lấy ví dụ như *if (Độ - độ {a[m,n]} ≥ θ)* chẳng hạn). Mặc dù vậy, thuật toán trên đây minh họa rất rõ ràng về ý tưởng chính của kỹ thuật lấy ngưỡng.

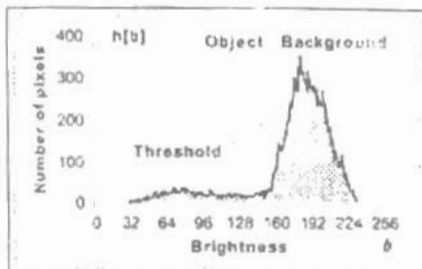
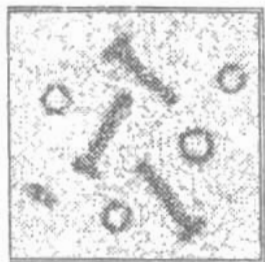
Câu hỏi trung tâm trong kỹ thuật lấy ngưỡng khi đó sẽ là: Chúng ta nên chọn ngưỡng θ như thế nào? Mặc dù không có thuật toán chọn ngưỡng vạn năng nào có thể áp dụng cho mọi loại ảnh, chúng ta cũng đã có rất nhiều phương pháp được đưa ra dưới đây.

Ngưỡng cố định

Phương pháp đầu tiên là chọn một ngưỡng độc lập với dữ liệu ảnh. Nếu chúng ta biết trước là chương trình ứng dụng sẽ làm việc với những ảnh có độ tương phản rất cao, trong đó các đối tượng quan tâm rất tối còn nền gần như đồng nhất (xem mục 7.1) và rất sáng, thì giá trị ngưỡng không đổi 128 trên thang độ sáng từ 0 đến 255 sẽ là một giá trị chọn khá chính xác. Chính xác ở đây nên được hiểu theo nghĩa là số lượng các điểm ảnh bị phân lớp sai là cực tiểu.

Ngưỡng dựa trên lược đồ

Trong hầu hết các trường hợp, ngưỡng được chọn từ lược đồ độ sáng của vùng hay ảnh cần được phân đoạn (xem mục 6.5.2 và 6.7). Hình 57 cho chúng ta một ví dụ về ảnh và lược đồ độ sáng liên kết với nó.



(a) Ảnh cần lấy ngưỡng (b) Lược đồ độ sáng của ảnh

Hình 52: Các điểm ảnh bên dưới ngưỡng ($a[m,n] < \theta$) sẽ được đánh nhãn là các điểm ảnh thuộc đối tượng. Những điểm ảnh ở trên ngưỡng sẽ được xem là những điểm ảnh nền.

Có rất nhiều kỹ thuật chọn ngưỡng tự động xuất phát từ lược đồ xám $\{h[b] / b = 0, 1, \dots, 2^B - 1\}$ đã được đưa ra. Những kỹ thuật phổ biến nhất trong số đó sẽ được trình bày dưới đây. Những kỹ thuật này có thể tận dụng những lợi thế do sự làm trơn dữ liệu lược đồ ban đầu mang lại nhằm loại bỏ những dao động nhỏ về độ sáng. Tuy nhiên, các thuật toán làm trơn cần phải cẩn trọng, không được làm dịch chuyển các vị trí đỉnh của lược đồ. Nhận xét này dẫn đến thuật toán làm trơn dưới đây, với giá trị W thông dụng là 3 hoặc 5:

$$h_{\text{smooth}}[b] = \frac{1}{W} \sum_{w=-(W-1)/2}^{(W-1)/2} h_{\text{raw}}[b-w] \quad W \text{ lẻ} \quad (199)$$

Kỹ thuật chọn ngưỡng theo kiểu lặp này do Ridler và Calvard đưa ra [35]. Trước hết lược đồ sẽ được phân đoạn thành hai phần bằng một giá trị ngưỡng khởi động như $\theta_0 = 2^{g-1}$, tức là bằng phân nửa thang độ xám động của ảnh. Sau đó, các trung bình mẫu ($m_{f,0}$) của những điểm ảnh thuộc đối tượng và ($m_{b,0}$) của những điểm ảnh nền sẽ được tính toán. Một giá trị ngưỡng mới θ_1 sẽ được tính kế đó bằng cách lấy giá trị trung bình của hai trung bình mẫu nói trên. Quá trình này cứ thế sẽ được tiếp tục với các ngưỡng mới cho đến khi nào giá trị ngưỡng không thay đổi nữa thì dừng lại. Nếu biểu diễn dưới dạng công thức toán học, chúng ta có:

$$\theta_k = \frac{m_{f,k-1} + m_{b,k-1}}{2} \quad \text{until } \theta_k = \theta_{k-1} \quad (200)$$

Thuật toán đối xứng nền

Kỹ thuật này dựa trên sự giả định là tồn tại hai đỉnh phân biệt trong lược đồ nằm đối xứng nhau qua đỉnh có giá trị lớn nhất trong phần lược đồ thuộc về các điểm ảnh nền. Kỹ thuật này có thể tận dụng các ưu điểm của việc làm trơn như đã mô tả trong phương trình (199). Đỉnh cực đại *maxp* tìm được nhờ tiến hành tìm giá trị cực đại trong lược đồ. Sau đó thuật toán sẽ được áp dụng tiếp ở phía *không phải là điểm ảnh thuộc đối tượng* ứng với giá trị cực đại đó nhằm tìm ra điểm *p%* như trong phương trình (71).

Trong hình 52(b), các điểm ảnh thuộc đối tượng nằm về bên trái của đỉnh nền (tại vị trí 183), điều này có nghĩa là cần tìm kiếm ở bên phải đỉnh đó để xác định được một giá trị ví dụ như 95%. Tại giá trị sáng này, 5% số điểm ảnh sẽ ở về phía bên phải (có giá trị sáng lớn hơn) nó. Tính hướng này xảy ra tại vị trí 216 trong hình 52(b). Do tính đối xứng đã giả

định ở trên, chúng ta sử dụng độ dịch chuyển về phía trái của điểm cực đại làm giá trị ngưỡng. Giá trị này bằng với độ dịch chuyển về phía phải của điểm cực đại đến nơi $p\%$ được tìm thấy. Với hình 52(b) chúng ta có giá trị ngưỡng cụ thể là $183 - (216 - 183) = 150$. Công thức tổng quát cho giá trị này sẽ là:

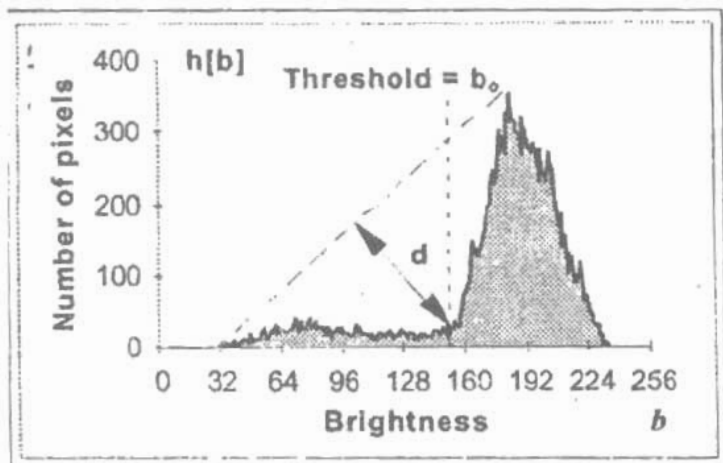
$$\theta = \max p - (p\% - \max p) \quad (201)$$

Kỹ thuật này dễ dàng điều chỉnh được cho phù hợp với tình huống ảnh có các đối tượng sáng trên một nền tối và tối. Ngoài ra, chúng ta còn có thể sử dụng nó khi đỉnh của đối tượng là trội hơn, và có thể giả sử được là phân bố độ sáng quanh đỉnh đối tượng là có tính đối xứng. Một biến thể khác của phương pháp này lại tận dụng cách ước lượng độ lệch chuẩn của mẫu (xem phương trình (69)) dựa vào một phía của đỉnh trội, sau đó sử dụng ngưỡng $\theta = \max p \pm 1.96s$ (ở mức 5%) hoặc $\theta = \max p \pm 2.57s$ (ở mức 1%). Sự lựa chọn dấu “+” hay “-” phụ thuộc vào hướng để xác định $\max p$ là ngưỡng nền/ đối tượng. Trong phương pháp này, các phân phối nên có dạng xấp xỉ phân phối Gauss quanh giá trị $\max p$, khi đó các giá trị 1.96 và 2.57 trong thực tế sẽ tương ứng với các mức 5% và 1%.

Thuật toán tam giác

Thuật toán này do Zack đưa ra trong [36] và được minh họa trong hình 53. Trong hình này, chúng ta có thể quan sát thấy một đường thẳng đã được xây dựng bằng cách nối từ giá trị lớn nhất của lược đồ tại độ sáng b_{\max} đến giá trị nhỏ nhất của lược đồ tại độ sáng b_{\min} . Với mỗi độ sáng b trong khoảng $[b_{\max}, b_{\min}]$, chúng ta đi tính khoảng cách d từ giá trị lược đồ tại b và $h[b]$ đến đường thẳng đã có. Giá trị b_0 ứng với

khoảng cách d lớn nhất sẽ được chọn làm giá trị ngưỡng θ . Kỹ thuật này đặc biệt hiệu quả khi các điểm ảnh thuộc đối tượng tạo nên một đỉnh yếu trong lược đồ ảnh.



Hình 53. Thuật toán tam giác dựa trên việc tìm ra giá trị độ sáng b cho khoảng cách d lớn nhất

Trong ba thuật toán nói trên, ứng với ảnh trong hình 52(a), chúng ta có $\theta = 139$ cho trường hợp thuật toán đẳng liệu, $\theta = 150$ cho trường hợp thuật toán đối xứng nền ở mức 5%, và $\theta = 152$ cho trường hợp thuật toán tam giác. Kỹ thuật lấy ngưỡng không nhất thiết phải được áp dụng cho toàn bộ ảnh, mà có thể áp dụng cho từng vùng ảnh một. Trong [37], hai tác giả Chow và Kaneko đã phát triển một biến thể của kỹ thuật lấy ngưỡng bằng cách chia một ảnh có kích thước $M \times N$ ra thành nhiều vùng không chồng lấp lên nhau. Các giá trị ngưỡng được tính riêng biệt cho từng vùng một và sau đó được kết hợp lại thông qua phép nội suy để hình thành nên một mặt ngưỡng cho toàn bộ ảnh. Trong thuật toán mới này, kích thước của các vùng cần được chọn một cách thích

hợp sao cho có một lượng đáng kể các điểm ảnh ở trong một vùng, nhằm phục vụ cho việc tính lược đồ và xác định ngưỡng tương ứng. Tính hữu ích của thuật toán này, cũng như nhiều thuật toán khác, sẽ phụ thuộc vào từng ứng dụng cụ thể.

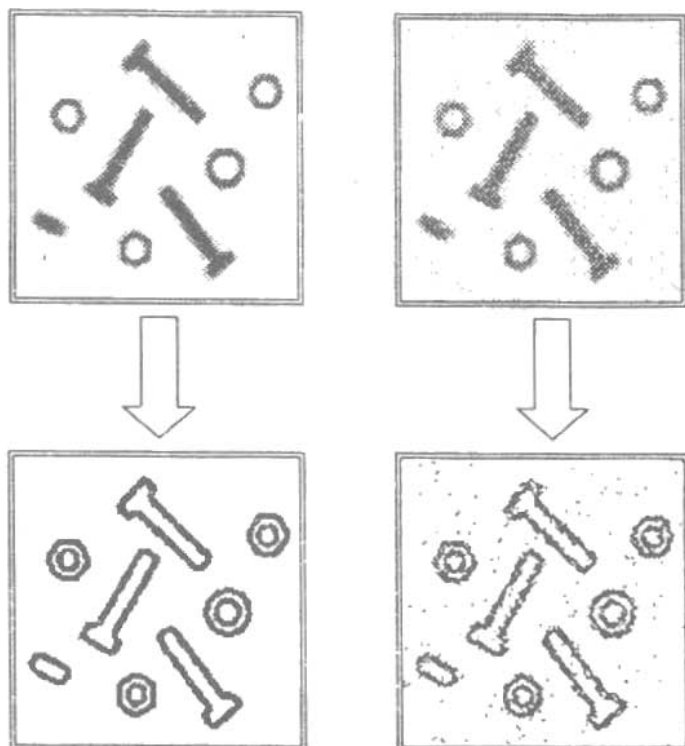
7.3.2. Kỹ thuật dò cạnh

Kỹ thuật lấy ngưỡng tạo ra sự phân đoạn ảnh bằng cách phân loại các điểm ảnh vào một trong hai nhóm đối tượng hoặc nền. Chúng ta còn có một cách khác để phân đoạn dựa vào việc tìm các điểm thuộc biên của những đối tượng khác nhau trong ảnh. Nhóm kỹ thuật này được gọi là các kỹ thuật dò cạnh. Dựa vào phần trình bày về lý thuyết morphology ở mục 6.5.10, đặc biệt là trên cơ sở phương trình (79), (163), và (170), chúng ta nhận thấy có một mối quan hệ mật thiết giữa hai khái niệm biên và vùng.

Các thuật toán dò cạnh dựa trên gradient

Điểm mấu chốt trong các kỹ thuật dò cạnh là tìm ra những thủ tục cho phép tạo các biên *đóng* quanh những đối tượng cần quan tâm. Với những đối tượng có SNR cao, các biên này có thể được tạo ra nhờ việc tính gradient và sau đó sử dụng một ngưỡng thích hợp. Hình 54 minh họa cho ý tưởng này.

Trong khi kỹ thuật này làm việc tốt với ảnh 30dB trong hình 54(a), thì nó lại không thể xác định chính xác những điểm ảnh nằm trên biên của các đối tượng trong ảnh 20dB ở hình 54(b). Có rất nhiều kỹ thuật làm trơn được trình bày trong mục 6.10 và trong phương trình (181) mà chúng ta có thể áp dụng để giảm tác động của nhiễu trước khi cho thực hiện toán tử gradient.

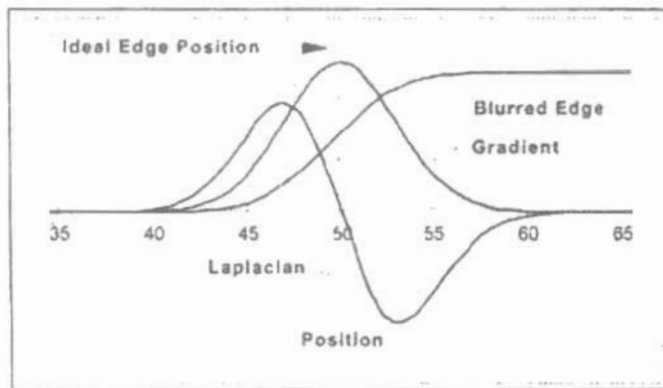


Hình 54. *Dò cạnh dựa trên gradient Sobel được mô tả trong phương trình (110), kết hợp với thuật toán lấy ngưỡng dạng liểu ở phương trình (92)*

Các thuật toán dựa trên giao điểm với trục hoành

Có một cách nhìn hiện đại hơn để giải quyết bài toán cạnh trong các ảnh có nhiễu là tìm giao điểm với trục hoành của toán tử Laplace ứng với mỗi ảnh (xem mục 6.11.2). Cơ sở của phương pháp này bắt nguồn từ một mô hình cạnh lý tưởng, là một hàm bước, bị làm mờ đi bởi một *OTF* như trong bảng 11 - T.3 (ngoài tiêu cự), 11 - T.5 (nhiều có giới hạn),

hoặc 11 – T.6 (mô hình tổng quát) để cho ra kết quả trong hình 55.



Hình 55. *Dò cạnh dựa trên nghiệm của đạo hàm bậc hai, hay còn gọi là toán tử Laplace. Các đường cong ở đây không được hiển thị theo đúng tỷ lệ.*

Theo mô hình này, vị trí cạnh là chỗ trong ảnh có toán tử Laplace đổi dấu, hay nói cách khác là tại giao điểm của nó với trục hoành. Do phép toán Laplace này bao gồm cả đạo hàm bậc hai, nên điều này có nghĩa là có một lượng nhiều đáng kể tại những tần số tọa độ cao (xem phương trình (114)). Để ngăn chặn nhiễu này, chúng ta cần phải sử dụng thêm kỹ thuật làm trơn.

Một bộ lọc trơn thích hợp trong số nhiều bộ lọc có thể có như đã được mô tả trong mục 6.10, nên tuân theo những tính chất sau do Canny đưa ra:

- Trong miền tần số, (u,v) hay (Ω, Ψ) bộ lọc càng hẹp càng tốt để có thể loại bỏ nhiễu có tần số cao.
- Trong miền tọa độ, (x,y) hay $[m,n]$, bộ lọc càng hẹp càng tốt nhằm cung cấp tính địa phương hóa tốt cho

cạnh. Một bộ lọc quá rộng có thể tạo nên sự bất định trong vùng có cạnh hiện hữu.

Bộ lọc trơn đồng thời thỏa mãn được cả hai tính chất nói trên – có bề rộng dải tần và bề rộng tọa độ là nhỏ nhất – chính là bộ lọc Gauss đã mô tả trong mục 6.10. Điều này có nghĩa là ảnh nên được làm trơn với một bộ lọc Gauss và theo sau là sự áp dụng phép toán Laplace. Biểu diễn bằng công thức chúng ta có:

$$\text{ZeroCrossing}\{a(x,y)\} = \{(x,y) | \nabla^2\{g_{2D}(x,y) \otimes a(x,y)\} = 0\} \quad (202)$$

trong đó $g_{2D}(x,y)$ được xác định bởi phương trình (93). Phép toán đạo hàm có tính tuyến tính và bất biến dịch chuyển như đã được định nghĩa trong hai phương trình (85) và (86). Điều này có nghĩa là bậc của các toán tử có thể hoán đổi được (36) hoặc có thể kết hợp lại thành một bộ lọc duy nhất (phương trình (37)). Cách tiếp cận thứ hai này dẫn đến công thức Marr–Hildreth của bộ lọc “Laplace của Gasuu” (LoG) xác định như sau [39].

$$\text{ZeroCros sin } g\{a(x,y)\} = \{(x,y) | LoG(x,y) \otimes a(x,y) = 0\} \quad (203)$$

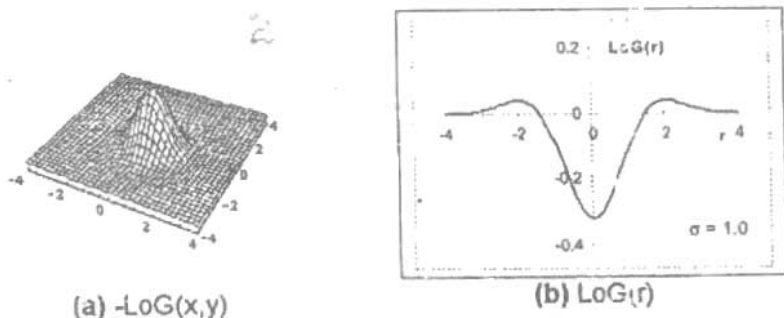
Trong đó:

$$LoG(x,y) = \frac{x^2 + y^2}{\sigma^4} g_{2D}(x,y) - \frac{2}{\sigma^2} g_{2D}(x,y) \quad (204)$$

Cho trước tính đối xứng vòng thì công thức trên có thể được viết lại thành:

$$LoG(r) = \left(\frac{r^2 - 2\sigma^2}{2\pi\sigma^6} \right) e^{-\frac{r^2}{2\sigma^2}} \quad (205)$$

Nhân chập hai chiều này đôi khi được gọi là bộ lọc “nón Mehico”, có dạng như trong hình 56.



Hình 56: Bộ lọc LoG với $\sigma = 1.0$

Các thuật toán dựa trên bộ lọc PLUS

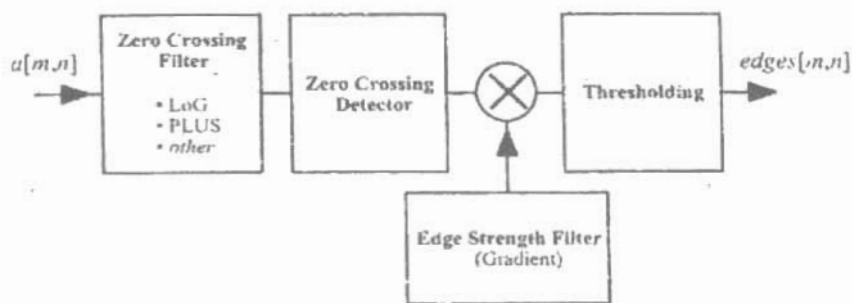
Trong số các thủ tục dò cạnh dựa trên nghiệm của đạo hàm bậc hai, có lẽ thủ tục cho kết quả chính xác nhất là bộ lọc PLUS do Verbeek và Van Vliet phát triển [40]. Bộ lọc này được xác định nhờ các phương trình (121) và (122) như sau:

$$\begin{aligned}
 PLUS(a) &= SDGD(a) + Laplace(a) \\
 &= \frac{A_{xx}A_x^2 + 2A_{xy}A_xA_y + A_{yy}A_y^2}{A_x^2 + A_y^2} + (A_{xx} + A_{yy}) \quad (206)
 \end{aligned}$$

Các chủ đề về tính chất đạo hàm của PLUS cũng như việc đánh giá tính chính xác của nó, mặc dù hết sức lý thú, lại vượt quá khuôn khổ của quyển sách này. Để đề nói rằng, với những cạnh cong tích cực trong các ảnh xám, thủ tục này ước lượng dư vị trí cạnh, còn với thủ tục dựa trên SDGD, vị trí cạnh lại bị ước lượng thiếu. Điều này đúng trong cả các ảnh 2-chiều và 3-chiều với bậc của sai số vào khoảng $(\sigma/R)^2$ trong đó R là bán kính cong của cạnh. Toán tử PLUS có bậc

của sai số vào khoảng $(\sigma/R)^4$ nếu ảnh được lấy mẫu với tần số tối thiểu, bằng 3 lần tần số lấy mẫu Nyquist thông thường như trong phương trình (56), hoặc nếu chúng ta chọn $\sigma \geq 2.7$ và lấy mẫu với tần số Nyquist thông thường.

Tất cả các phương pháp dò cạnh dựa trên việc tìm nghiệm của toán tử Laplace đều phải phân biệt rõ ràng hai khái niệm *điểm cắt zero* và *trị zero*. Trong khi khái niệm trước thể hiện các vị trí cạnh, thì khái niệm sau được sinh ra bởi những vùng không phức tạp hơn các mặt song tuyến tính, tức là các mặt được xác định bởi quan hệ $a(x,y) = a_0 + a_1 \cdot x + a_2 \cdot y + a_3 \cdot x \cdot y$. Để phân biệt hai tình huống này, chúng ta trước tiên sẽ đi tìm các vị trí điểm cắt zero và gán cho chúng nhãn "1" còn các điểm ảnh khác có nhãn "0". Sau đó chúng ta nhân ảnh kết quả với số đo *độ dài cạnh* tại mỗi điểm ảnh. Có rất nhiều độ đo độ dài cạnh dựa trên khái niệm gradient, như đã được mô tả trong mục 6.11.1 và phương trình (182). Trong phương trình (182), chúng ta có một độ đo gradient theo kiểu morphology cho độ dài cạnh rất hiệu quả, do Lee, Aralick, và Shapiro đưa ra trong [41]. Sau khi nhân, ảnh sẽ được lấy

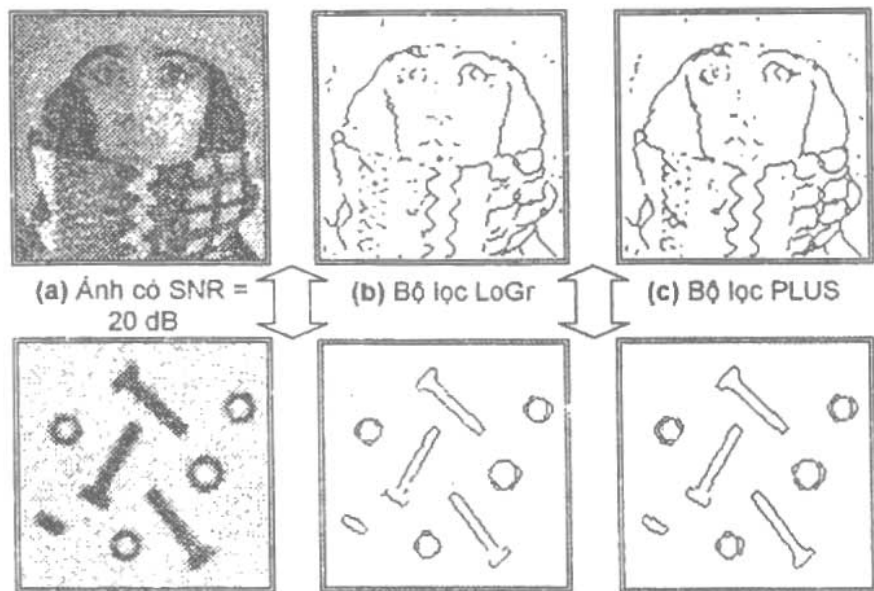


Hình 57. Chiến lược tổng quát để dò cạnh theo kiểu các điểm cắt zero

ngưỡng như đã trình bày ở trên để cho ra kết quả sau cùng. Thủ tục này do vậy có thể minh họa bởi hình 57 [42].

Kết quả của hai kỹ thuật dò cạnh dựa trên điểm cắt zero, là bộ lọc *LoG* và bộ lọc *PLUS*, được chỉ ra trong hình 58 với ảnh có *SNR* và 20dB.

Các kỹ thuật dò cạnh nói trên mang lại cho chúng ta một ảnh gồm toàn các điểm cạnh. Để quyết định xem chúng thuộc về các đối tượng, hay thuộc về những đường thẳng đơn giản trong một ảnh, chúng ta còn phải áp dụng thêm các kỹ thuật lấp đầy vùng như đã được giới thiệu, ví dụ như theo phương trình (170).



Hình 58. Dò cạnh nhờ các thuật toán điểm cắt zero *LoG* và *PUS*. Trong cả hai thuật toán chúng ta lấy $\sigma = 1.5$

7.3.3. Kỹ thuật morphology nhị phân

Những thuật toán mà chúng ta đã trình bày về lý thuyết morphology trong mục 6.10 có thể kết hợp lại với nhau để cho ra những kỹ thuật xử lý ảnh nhị phân và ảnh xám hết sức mạnh mẽ. Do ảnh nhị phân thường là kết quả phân đoạn của một ảnh xám, nên sự xử lý theo kiểu morphology trên ảnh nhị phân kết quả có thể giúp cải thiện chất lượng phân đoạn trước đó rất nhiều.

Lọc theo kiểu muối-hoặc-tiêu

Các thủ tục phân đoạn thường cho ra các điểm ảnh bị cô lập "1" trong một lân cận "0" (muối) hoặc cho ra các điểm ảnh bị cô lập "0" trong một lân cận "1" (tiêu). Định nghĩa lân cận phù hợp cho từng tình huống cần phải được chọn như trong hình 4. Bằng cách sử dụng công thức bảng tra cho các phép toán Boolean trên một lân cận 3×3 như đã được mô tả trong hình 44, các bộ lọc muối và bộ lọc tiêu sẽ được cài đặt hết sức dễ dàng. Chúng ta đánh trọng số các vị trí khác nhau trong lân cận 3×3 như sau:

$$\text{Weights} = \begin{bmatrix} w_4 = 16 & w_3 = 8 & w_2 = 4 \\ w_5 = 32 & w_0 = 1 & w_1 = 2 \\ w_6 = 64 & w_7 = 128 & w_8 = 256 \end{bmatrix} \quad (207)$$

Với một cửa sổ 3×3 trong $a[m,n]$, với các trị "0" hay "1" chúng ta sẽ tính:

$$\begin{aligned} \text{sum} = & w_0 a[m,n] + w_1 a[m+1,n] + w_2 a[m+1,n-1] + \\ & w_3 a[m,n-1] + w_4 a[m-1,n-1] + w_5 a[m-1,n] + \\ & w_6 a[m-1,n+1] + w_7 a[m,n+1] + w_8 a[m+1,n-1] \end{aligned} \quad (208)$$

Giá trị kết quả, *sum*, là một đại lượng bị chặn trong khoảng $0 \leq \text{sum} \leq 511$.

Bộ lọc muối

Các phiên bản liên thông 4 và liên thông 8 của bộ lọc này là giống nhau và được cho bởi công thức sau:

i) *Tính sum*

$$\begin{array}{ll} \text{ii) If (sum = 1)} & c[m,n] = 0 \\ \text{Else} & c[m,n] = a[m,n] \end{array} \quad (209)$$

Bộ lọc tiêu

Các phiên bản liên thông 4 và liên thông 8 của bộ lọc này được cho bởi các công thức sau:

<i>Liên thông 4</i>	<i>Liên thông 8</i>	
i) <i>Tính sum</i>	i) <i>Tính sum</i>	
ii) <i>If (sum == 170)</i>	ii) <i>If (sum == 510)</i>	
$c[m,n] = 1$	$c[m,n] = 1$	(210)
<i>Else</i>	<i>Else</i>	
$c[m,n] = a[m,n]$	$c[m,n] = a[m,n]$	

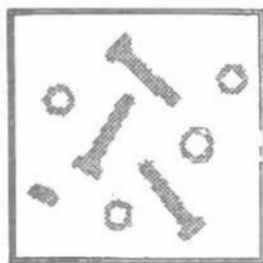
Cô lập các đối tượng có lỗ

Để tìm những đối tượng có lỗ, chúng ta sử dụng thủ tục sau đây, như được minh họa trong hình 59.

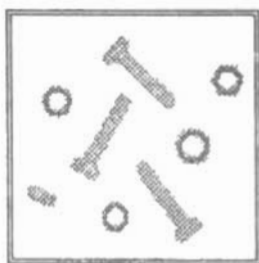
- i) *Phân đoạn ảnh* để tạo ra biểu diễn mặt nạ nhị phân.
- ii) *Tính xương* không có các điểm cuối theo phương trình (169).
- iii) *Sử dụng bộ lọc muối* để loại bỏ các điểm xương đơn.

iv) *Lan truyền* các điểm xương còn lại vào trong mặt nạ nhị phân ban đầu theo phương trình (170). (211)

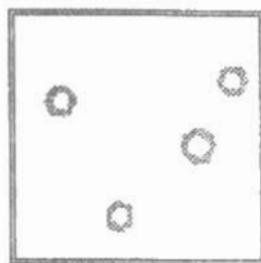
Trong hình vẽ này, các đối tượng nhị phân có màu xám còn các xương sau khi áp dụng bộ lọc muối sẽ có màu đen. Chúng ta cần chú ý rằng thủ tục này hoàn toàn không sử dụng thêm tham số nào khác ngoài sự lựa chọn cơ sở để tính liên thông. Trong ví dụ ở hình 59, định nghĩa liên thông 8 được sử dụng cấu trúc $B = N_8$.



(a) Ảnh nhị phân



(b) Xương sau khi lọc muối



(c) Các đối tượng có lỗ

Hình 59. *Cô lập các đối tượng có lỗ bằng các phép toán morphology*

Lấp đầy các lỗ trong các đối tượng

Để lấp đầy các lỗ trong các đối tượng, chúng ta sử dụng thủ tục sau đây, như được minh họa trong hình 60.

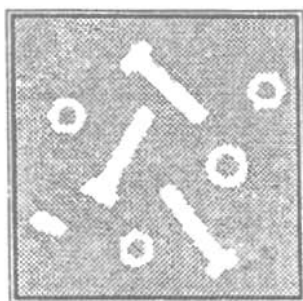
i) *Phân đoạn ảnh* để tạo ra biểu diễn nhị phân của các đối tượng.

ii) Tính *phần bù* của ảnh nhị phân và lấy đó làm ảnh mặt nạ.

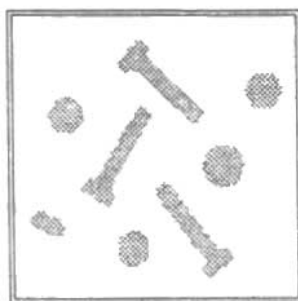
iii) Sinh ra một *ảnh hạt giống* làm biên của ảnh ban đầu.

iv) Lan truyền ảnh hạt giống vào mặt nạ (phương trình (97).

v) Lấy phần bù của kết quả lan truyền nhằm tạo ra kết quả sau cùng. (212)



(a) Ảnh mặt nạ và ảnh hạt giống



(b) Các đối tượng có lỗ đã được lấp đầy

Hình 60. Lấp đầy lỗ trong đối tượng

Trong hình 60(a), ảnh mặt nạ có màu xám còn ảnh hạt giống có màu đen. Khi những điểm ảnh thuộc đối tượng được đặc tả với tính liên thông $C = 8$, thì sự lan truyền vào ảnh mặt nạ (nền ảnh) cần được thực hiện với tính liên thông $C = 4$, tức là được phát triển với phần tử cấu trúc $B = N_4$.

Loại bỏ đối tượng dính biên

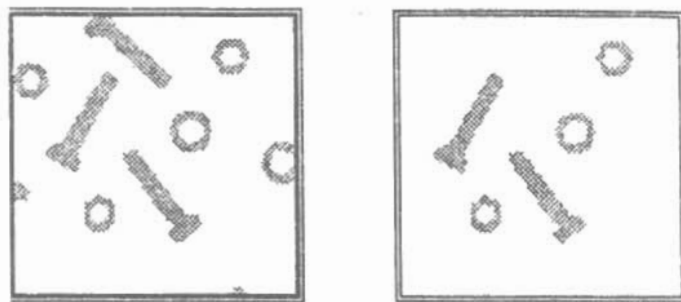
Các đối tượng liên thông với biên ảnh cần phải được loại bỏ do chúng không thích hợp cho việc phân tích ảnh. Để triệt tiêu chúng, chúng ta có thể sử dụng một loạt phép toán morphology được minh họa trong hình 61.

i) Phân đoạn để tạo ra ảnh mặt nạ nhị phân của các đối tượng.

ii) Sinh ra một ảnh hạt giống và lấy đó làm biên ảnh.

ii) Lan truyền hạt giống đó vào mặt nạ theo phương trình (97).

iv) Tính XOR của kết quả lan truyền với ảnh mặt nạ, đó chính là kết quả sau cùng.



(a) Ảnh mặt nạ và ảnh hạt giống (b) Các đối tượng còn lại

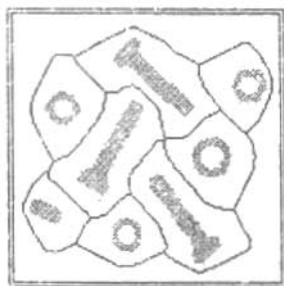
Hình 61. Loại bỏ các đối tượng dính biên

Trong hình 61(a), ảnh mặt nạ có màu xám còn ảnh hạt giống có màu đen. Nếu phần tử cấu trúc được sử dụng trong lan truyền là $B = N_4$, thì các đối tượng bị loại sẽ có liên thông 4 với biên ảnh. Nếu $B = N_8$, được sử dụng thì các đối tượng bị loại sẽ có liên thông 8 với biên ảnh.

Exo-skeleton

Exo-skeleton của một tập các đối tượng chính là xương của nền ảnh chứa các đối tượng đó. Xương này phân hoạch ảnh thành các vùng khác nhau, trong đó mỗi đối tượng thuộc về một vùng. Phép rút xương thật sự theo phương trình (169) sẽ được thực hiện và không bảo toàn các điểm ảnh cuối, đồng thời thiết lập giá trị biên là "0". Thủ tục này được mô tả ở phần dưới đây, còn kết quả thực hiện được minh họa trong hình 62.

- i) *Phân đoạn để tạo ra ảnh nhị phân.*
- ii) *Tính phần bù của ảnh nhị phân đó.*
- iii) *Tính xương ảnh bằng cách sử dụng phương trình*
 $(169) \ i + ii$ *với biên được cho giá trị "0".* (124)

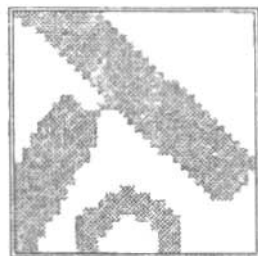


Hình 62. *Exo - skeleton*

Các đối tượng dính

Các thủ tục phân đoạn ảnh thường khó tách được các đối tượng khác biệt nhưng dính nhau trong ảnh. Sau đây là một phương pháp để tách những đối tượng như vậy. Như đã biết, các thủ tục *exo-skeleton* phân hoạch ảnh thành nhiều vùng, và mỗi vùng chứa một đối tượng. Phép rút xương thật sự sẽ được thực hiện và không bảo toàn các điểm ảnh cuối, đồng thời thiết lập giá trị biên là "0". Thủ tục này được minh họa trong hình 63.

- i) *Phân đoạn để tạo ra ảnh nhị phân.*
- ii) *Tính "một lượng nhỏ" các phép bào mòn với $B = N_4$.*
- iii) *Tính *exo-skeleton* của phần đã bào mòn.*
- iv) *Lấy phần bù của kết quả sau khi thực hiện *exo-skeleton*.*
- v) *Tính AND của ảnh nhị phân ban đầu và phần bù *exo-skeleton*.*



(a) Ảnh đã bào mòn và ảnh exo-skeleton

(b) Các đối tượng đã được tách (chi tiết)

Hình 63. Tách các đối tượng dính nhau

Trong hình 63(a), ảnh nhị phân đã bị bào mòn có màu xám còn ảnh exo-skeleton có màu đen. Hình 63(b) cho thấy một phần ảnh kết quả sau cùng được phóng lớn và chúng ta có thể quan sát thấy dễ dàng sự tách biệt giữa các đối tượng. Giá trị cụ thể của “một lượng nhỏ” các phép bào mòn được sử dụng trong thuật toán này chỉ cần đủ để tạo nên một sự tách thô ban đầu giữa các đối tượng. Nếu exo-skeleton có liên thông 8 thì phần nền tách các đối tượng của sẽ có liên thông 8. Bản thân các đối tượng sẽ được tách rời theo tiêu chuẩn liên thông 4 (xem mục 6.10 và hình 37).

7.3.4. Kỹ thuật morphology mức xám

Như chúng ta đã có dịp thấy trong mục 7.1.2, các kỹ thuật morphology mức xám rất hữu dụng trong một số bài toán, chẳng hạn như trong việc hiệu chỉnh sắc thái. Trong mục này, chúng ta sẽ tìm hiểu thêm một số kỹ thuật khác nữa.

Biến đổi chóp mũ

Sự cô lập những đối tượng xám hoàn toàn có thể thực hiện được nhờ vào phép biến đổi chóp mũ do Meyer đưa ra trong [43, 44]. Phụ thuộc vào tình huống chúng ta đang phải làm việc là với những đối tượng sáng trên nền tối hay đối

tượng tối trên nền sáng, mà phép biến đổi chóp mũ sẽ có một trong hai dạng sau:

Các đối tượng sáng -

$$TopHat(A, B) = A - (A \circ B) = A - \max_B \left(\min(A) \right) \quad (216)$$

Các đối tượng tối -

$$TopHat(A, B) = A \bullet B - A \min_B \left(\max(A) \right) - A \quad (217)$$

Trong đó phần tử cấu trúc B được chọn cần phải lớn hơn các đối tượng cần cô lập, và nếu có thể, nên có hình dạng lỗi. Do những tính chất đã cho trước trong các phương trình (155) và (158), $TopHat(A, B) \geq 0$. Hình 64 cho chúng ta một ví dụ minh họa về kỹ thuật này.

Trong hình 64, ảnh ban đầu bao gồm cả sắc thái được xử lý bởi một phần tử cấu trúc dạng 15×1 như đã mô tả trong các phương trình (216) và (217) để cho ra được kết quả mong muốn. Chúng ta cần lưu ý là phép biến đổi ứng với những đối tượng tối đã được xây dựng nhằm thu được các đối tượng "tích cực" thay vì các đối tượng "không tích cực". Những định nghĩa khác dĩ nhiên cũng có thể sử dụng ở đây.

Kỹ thuật lấy ngưỡng

Từ kỹ thuật xử lý theo kiểu morphology, chúng ta có thể đi đến một ước lượng đơn giản cho một mặt ngưỡng biến đổi cục bộ như sau:

$$\text{Mặt ngưỡng} - \theta(m, n) = \frac{1}{2} (\max(A) + \min(A)) \quad (218)$$

Một lần nữa, chúng ta lại thay phần tử cấu trúc B bằng hai phép toán max và min nhằm giúp cho việc thể hiện thuật toán đơn giản hơn về mặt ký hiệu.

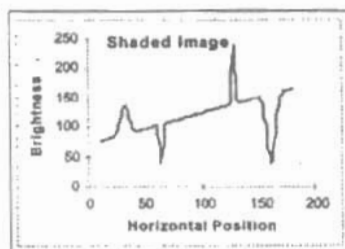
Co giãn độ tương phản cục bộ

Bằng cách sử dụng các phép toán morphology, chúng ta có thể cài đặt được kỹ thuật *co giãn độ tương phản cục bộ*. Tức là, lượng co giãn sẽ được áp dụng cho một lân cận hoàn toàn phụ thuộc vào độ tương phản ban đầu trong lân cận đó. Để đo độ tương phản cục bộ trong cửa sổ xác định bởi phần tử cấu trúc B , chúng ta có thể sử dụng toán tử gradient theo kiểu morphology như đã được định nghĩa trong phương trình (182):

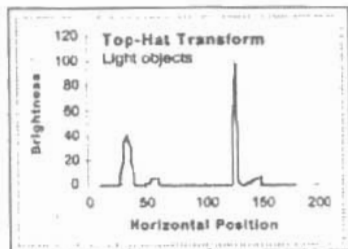
$$LocalContrast(A, B) = \max(A) - \min(A) \quad (219)$$

Thủ tục co giãn độ tương phản cục bộ sẽ được cho bởi:

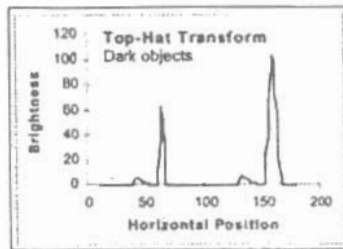
$$c[m, n] = \text{scale} \cdot \frac{A - \min(A)}{\max(A) - \min(A)} \quad (220)$$



(a) Ảnh ban đầu



(b) Biến đổi cho đối tượng sáng



(c) Biến đổi cho đối tượng tối

Hình 64. Các biến đổi chóp - mũ

Ở đây hai phép toán *max* và *min* được áp dụng trên phần tử cấu trúc B. Hình 65 cho chúng ta tác động của thủ tục này trên ảnh. Chúng ta có thể quan sát thấy phép toán cục bộ này là một phiên bản mở rộng của phép toán *điểm* dùng để co giãn độ tương phản vốn đã được định nghĩa trong phương trình (77).

Bằng cách sử dụng các ảnh kiểm tra chuẩn (như chúng ta đã có dịp quan sát thấy trong nhiều ví dụ), chúng ta có thể minh họa sức mạnh của cách tiếp cận lọc theo kiểu morphology cục bộ này.



Hình 65. Co giãn độ tương phản cục bộ

TÀI LIỆU THAM KHẢO

1. Dudgeon, D.E and R.M. Mersereau, *Multidimensional Digital Signal Processing*, 1984, Englewood Cliffs, New Jersey: Prentice-Hall.
2. Castleman, K.R., *Digital Image Processing*, Second ed, 1996, Englewood Cliffs, New Jersey: Prentice-Hall.
3. Oppenheim, A.V., A.S. Willsky, and I. Young, *Systems and Signals*, 1983, Englewood Cliffs, New Jersey: Prentice-Hall.
4. Papoulis, A., *Systems and Transforms with Applications in Optics*, 1986, New York: McGraw-Hill.
5. Russ, J.C., *The Image Processing Handbook*, Second ed, 1995, Boca Raton, Florida: CRC Press.
6. Giardina, C.R. and E.R. Dougherty, *Morphological Methods in Image and Signal Processing*, 1988, Englewood Cliffs, New Jersey: Prentice-Hall, 321.
7. Gonzalez, R.C. and R.E. Woods, *Digital Image Processing*, 1992, Reading, Massachusetts. Addison-Wesley. 761.
8. Goodman, J.W., *Introduction to Fourier Optics*, McGraw-Hill Physical and Quantum Electronics Series, 1968, New York: McGraw - Hill, 287.
9. Heijmans, H.J.A.M, *Morphological Image Operators - Advances in Electronics and Electron Physics*, 1994, Boston: Academic Press.
10. Hunt, R.W.G., *The Reproduction of Colour in Photography, Printing & Television*, Fourth ed, 1987, Tolworth, England: Fountain Press.

11. Freeman, H.m, *Boundary encoding and processing, in Picture Processing and Psychopictorics*, B.S. Lipkin and A. Rosenfeld, Editors, 1970, Academic Press: New York. P. 241–266.
12. Stockham, T.G., *Image Processing in the Context of a Visual Model, Proc.IEEE*, 1972, 60: p.828 – 824.
13. Murch, G.M., *Visual and Auditory Perception*, 1973, NewYork: Bobbs–Merrill Company, Inc. 403.
14. Frisby, J.P., *Seeing: Illusion, Brain and Mind*, 1980, Oxford, England: Oxford University Press, 160.
15. Blakemore, C. and F.W.C. Campbell, *On the existence of neurons in the human visual system selectively sensitive to the orientation and size of retinal images*, *J.Physiology*, 1969, 203: p.237–260.
16. Born, M. and E. Wolf, *Principles of Optics*, Sixth ed, 1980, Oxford: Pergamon Press.
17. Young, I.T., *Quantitative Microscopy*. *IEEE Engineering in Medicine and Biology*, 1996, 15 (1): p.59–66.
18. Dorst, L. and A.W.M. Smeulders, *Length estimators compared, in Pattern Recognition in Practice II*, E.S. Gelsema and L.N. Kanal, Editors, 1986, Elsevier Science: Amsterdam. P. 73–80.
19. Young, I.T., *Sampling density and quantitative microscopy*. *Analytical and Quantitative Cytology and Histology*, 1988. 10(4): p. 269–275.
20. Kulpa, Z., *Arcan and perimeter measurement of blobs in discrete binary pictures*, *Computer Vision, Graphics and Image Processing*, 1977. 6: p.434–454.

21. Vossepoel, A.M. and A.W.M. Smeulders, *Vector code probabilities and metrication error in the representation of straight lines of finite length*, Computer Graphics and Image Processing, 1982. 20: p.347-364.
22. Photometrics Ltd., *Signal Processing and Noise, in Series 2-CCD Cameras Manual*, 1990: Tucson, Arizona.
23. Huang, T.S., G.J. Yang, and G.Y. Tang, *A Fast Two - Dimensional Median Filtering Algorithm*, IEEE Transactions on Acoustics, Speech, and Signal Processing, 1979. ASSP-27: p.13-18.
24. Groen, F.C.A., R.J. Ekkers, and R. De Vries, *Image processing with personal computers*, Signal Processing, 1988. 15: p. 279-291.
25. Verbeek, P.W., H.A. Vrooman, and L.J. Van Vliet, *Low-Level Image Processing by Max-Min Filters*, Signal Processing, 1988. 15:p.249-258.
26. Toyng, I.T. and L.J. Van Vliet, *Recursive Implementation of the Gaussian Filter*, Signal Processing, 1995. 44(a): p. 139-151.
27. Kuwahara, M., et al. *Processing of RI-angiocardigraphic images, in Digital Processing of Biomedical Images*, K. Preston and M. Onoe, Editors, 1976, Plenum Press: New York. P.187-203.
28. Van Vliet, L.J., *Grey-scale measurements in multi-dimensional digitized images*, PhD Thesis: Delft University of Technology, 1993.
29. Serra, J. *Image Analysis and Mathematical Morphology*, 1982, London: Academic Press.

30. Vincent, L., *Morphological transformations of binary Images with arbitrary structuring elements*, Signal Processing, 1991. 22(1): p.3–23.
31. Van Vliet, L.J and B.J.H. Verwer, *A Contour Processing Method for Fast Binary Neighbourhood Operations*, Pattern Recognition Letters, 1988 7(1): P.27–36.
32. Young, I.T., et al., *A new implementation for the binary and Minkowski operators*, Computer Graphics and Image Processing, 1981. 17(3): p.189–210.
33. Lantúejoul, C., *Skeletonization in Quantitative Metallography, in Issues of Digital Image Processing*, R.M. Haralick and J.C. Simon,
34. Oppenheim, A.V, R.W. Schafer, and T.G. Stockham, Jr. *Non-Linear Filtering of Multiplied and Convolved Signals*, Proc. IEEE, 1968.56(8): p.1264–1291.
35. Ridler, T.W. and S. Calvard, *Picture thresholding using an iterative selection method*, IEEE Tran, on Systems, Man, and Cybernetics, 1978. SMC–8(8): p.530–632.
36. Zack, G.W., W.E. Rogers, and S.A.Latt, *Automatic Measurement Of Sister Chromatid Exchange Frequency*, 1977. 25(7): p.741–753/
37. Chow, C.K. and T. Kaneko, *Automatic boundary detection of the left ventricle from cineangiograms*, Computers and Biomedical Research, 1972. 5: p.388–410.
38. Canny, J.m, *A Computational Approach to Edge Detection*, IEEE Transactions on Pattern Analysis and Machine Intelligence, 1986. PAMI–8(6): p. 679–798.

39. Marr, D. and E.C.Hidreth, *Theory of edfsge detection*, Proc. R. Soc London Ser.B., 1980 207: p. 187–217.
40. Verbeek, P.V. and L.J. Van Vliet, *On the Location Error of Curved Edges in Low-Pass Filtered 2D and 3D Images*, IEEE Transactions on Pattern Analysis and Machine Intelligence, 1994. 16(7): p. 726–733.
41. Lee, J.S.L, R.M. Haralick, and L.S. Shapiro. *Morphologic Edge Detection. In 8th International Conference on Pattern Recognition, 1986*. Paris: IEEE Computer Society.
42. Van Vliet, L.J., I.T. Young, and A.L.D. Beckers, *A Non-linear Laplace Operator as Edge Detector in Noisy Images*, Computer Vision, Graphics, and Image Processing, 1989. 45: p.167–195.
43. Meyer, F. and S. Beucher, *Morphological Sementation*, *J.Visual Comm. Image Rep.*, 1990. 1(1): p.21–46.
44. Meyer, F., *Iterative Image Transformations for an Automatic Screening of Cevical Cancer*. Journal of Histochemistry and Cytochemistry, 1979. 27: p.128–135.

Giáo trình XỬ LÝ ẢNH

Võ Đức Khánh

NHÀ XUẤT BẢN

ĐẠI HỌC QUỐC GIA TP HỒ CHÍ MINH

KP 6, P. Linh Trung, Q. Thủ Đức, TPHCM

ĐT: 724 2181 + 1421, 1422, 1423, 1425, 1426

Fax: 724 2194; Email: vnuhp@vnuhcm.edu.vn

☆☆☆

Chịu trách nhiệm xuất bản

TS HUỖNH BÁ LÂN

Biên tập

NGUYỄN TIẾN NAM

Sửa bản in

PHẠM THỊ ANH TÚ

Trình bày bìa

GT .01. TH(V)
DHQG.HCM-07 81-46

TH.GT.965 – 07(T)

In 500 cuốn khổ 14,5 x 20,5cm tại Công ty in Hưng Phú. Số ĐKKHXB: 775-2008/CXB/134-54/ĐHQGTPHCM. Quyết định xuất bản số: 416/QĐ-ĐHQGTPHCM/TB ký ngày 22/8/2008. In xong và nộp lưu chiểu tháng 10/2008.

NHÀ SÁCH NGUYỄN VĂN CỬ



2 000103 009687

ISBN 978-990001-5482-2

000001 54822219

23,000 đ



Giá: 23.000đ