DỊCH MATLAB: 98-109

Chương 3 • Biến đổi cường độ và lọc không gian 98

trong đó các đối số được định nghĩa trước đây cho các đồ thị stem. Cũng như trong stem, các thuộc tính trong plot được chỉ định dưới dạng bộ ba. Mặc định cho plot là các đường màu xanh dương đậm không có dấu. Nếu một bộ ba được chỉ định mà giá trị ở giữa để trống (hoặc bỏ qua), thì không vẽ đường. Như trước đây, nếu bỏ qua horz, trục ngang được chia thành các đơn vị từ 0 đến độ dài (z).

Đồ thị trong Hình 3.7(d) được tạo ra bằng cách sử dụng các lệnh sau:

>> he = imhist(f);

>> plot(hc) % Sử dụng các giá trị mặc định.

>> axis([0 255 0 15000])

>> set(gca, 'xtick', [0:50:255])

>> set(gca, 'ytick', [0:2000:15000])

Hàm plot thường được sử dụng để hiển thị các hàm biến đổi (xem Ví dụ 3.5).

Trong thảo luận trước, các giới hạn trục và dấu tick được thiết lập thủ công. Để thiết lập các giới hạn và dấu tick tự động, sử dụng các hàm ylim và xlim, với mục đích của chúng ta ở đây, có các dạng cú pháp:

ylim('auto')

xlim('auto')

Trong số các biến thể cú pháp có thể có của hai hàm này (xem tài liệu trợ giúp để biết chi tiết), có một tùy chọn thủ công, được cho bởi:

ylim([ymin ymax])

xlim([xmin xmax])

cho phép thiết lập thủ công các giới hạn. Nếu các giới hạn được chỉ định chỉ cho một trục, thì các giới hạn trên trục còn lại sẽ được đặt thành 'auto' theo mặc định. Chúng ta sẽ sử dụng các hàm này trong phần tiếp theo. Gõ hold on tại dấu nhắc giữ lại đồ thị hiện tại và các thuộc tính trục nhất định để các lệnh vẽ đồ thị tiếp theo thêm vào đồ thị hiện có.

Một hàm vẽ đồ thị khác đặc biệt hữu ích khi xử lý các function handles (xem Mục 2.10.4 và 2.10.5) là hàm fplot. Cú pháp cơ bản là:

fplot(fhandle, limits, 'LineSpec')

trong đó fhandle là một function handle, và limits là một vector chỉ định giới hạn trục x, [xmin xmax]. Bạn sẽ nhớ từ thảo luận về hàm timeit trong Mục 2.10.5 rằng việc sử dụng các function

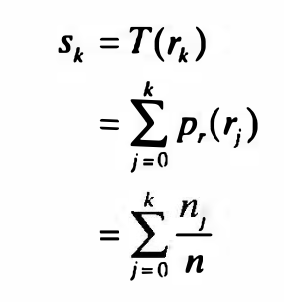
99

handles cho phép cú pháp của hàm cơ bản độc lập với các tham số của hàm cần xử lý (trong trường hợp này là vẽ đồ thị). Ví dụ, để vẽ hàm hyperbolic tangent, tanh, trong khoảng (-2 2] sử dụng đường chấm, chúng ta viết:

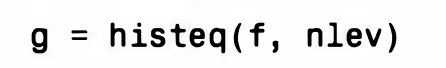
fplot(@tanh, [-2 2], ':')

**3.3.2 Cân Bằng Histogram**

Giả sử trong một khoảnh khắc rằng các mức cường độ là các giá trị liên tục được chuẩn hóa trong khoảng [0, 1], và gọi

​

(r) là hàm mật độ xác suất (PDF) của các mức cường độ trong một hình ảnh nhất định, trong đó chỉ số dưới được dùng để phân biệt giữa các PDF của hình ảnh đầu vào và đầu ra. Giả sử rằng chúng ta thực hiện phép biến đổi sau trên các mức đầu vào để thu được các mức cường độ đầu ra (đã xử lý),



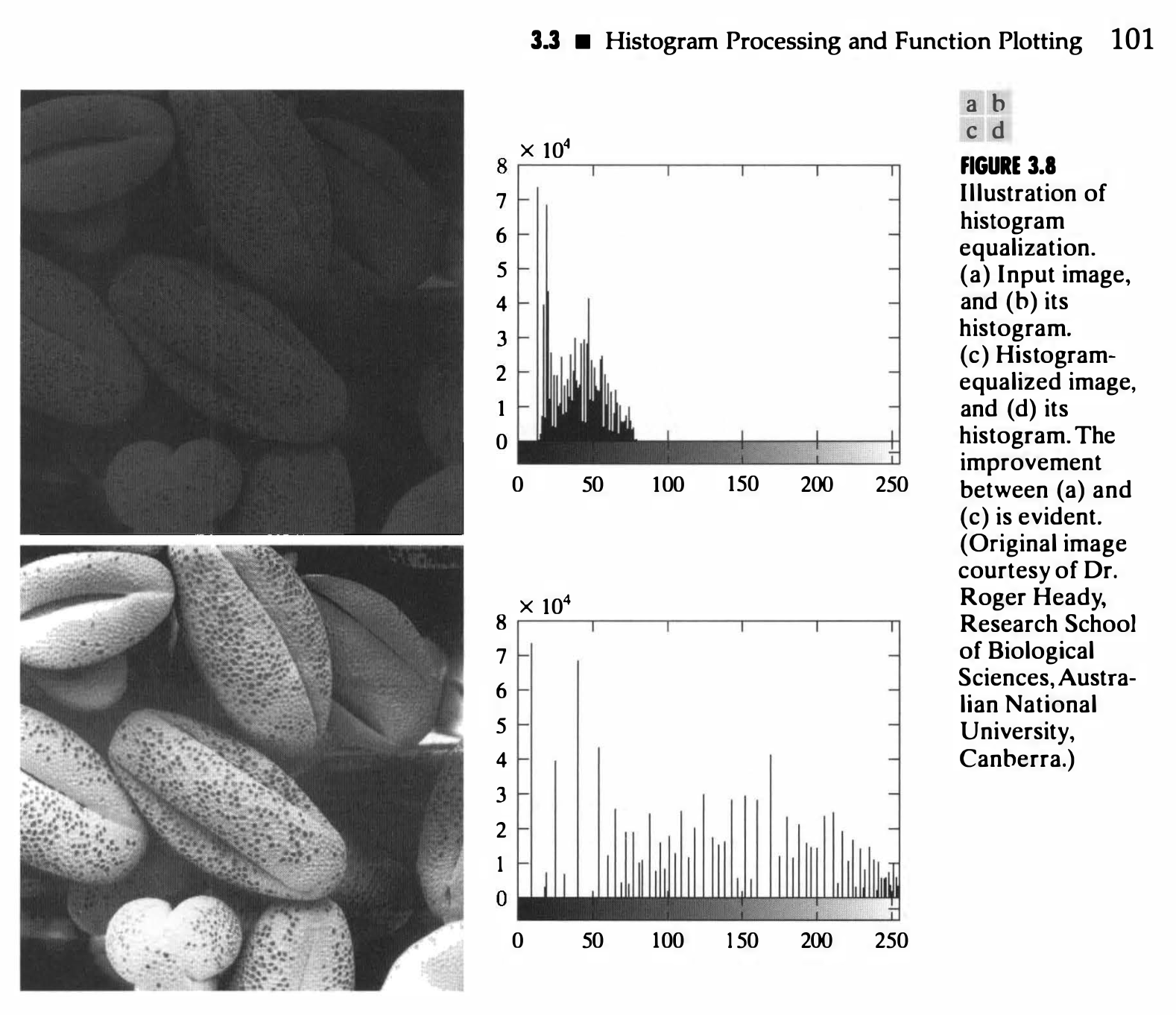
trong đó :

w là biến tích phân giả. Có thể chứng minh được (Gonzalez và Woods [2008]) rằng hàm mật độ xác suất của các mức đầu ra là đồng nhất

Hình 3.8(a) là ảnh chụp phấn hoa qua kính hiển vi điện tử, được phóng đại khoảng 700 lần. Về mặt cải tiến cần thiết, đặc điểm quan trọng nhất của hình ảnh này là nó tối và có dải động thấp. Những đặc điểm này được thể hiện rõ ràng trong biểu đồ ở Hình 3.8(b), trong đó bản chất tối của hình ảnh làm cho biểu đồ bị lệch về phía đầu tối của thang màu xám. Phạm vi động thấp được thể hiện rõ ràng từ thực tế là biểu đồ có hàng hẹp đối với toàn bộ thang màu xám. Đặt f biểu thị hình ảnh đầu vào, trình tự các bước sau đây sẽ tạo ra Hình. 3.8(a) đến (d):

>> imshow(f) ; % Fig. 3.8(a) .  
>> figure, imhist(f) % Fig. 3.8(b) . >> ylim('auto')  
>> g = histeq(f, 256);  
>> figure, imshow(g) % Fig. 3.S(c). >> figure, imhist(g) % Fig. 3.8(d) . >> ylim('auto')

Hình ảnh trong Hình 3.8(c) là kết quả đã được cân bằng biểu đồ. Sự cải thiện về cường độ và độ tương phản trung bình là điều hiển nhiên. Những tính năng này cũng



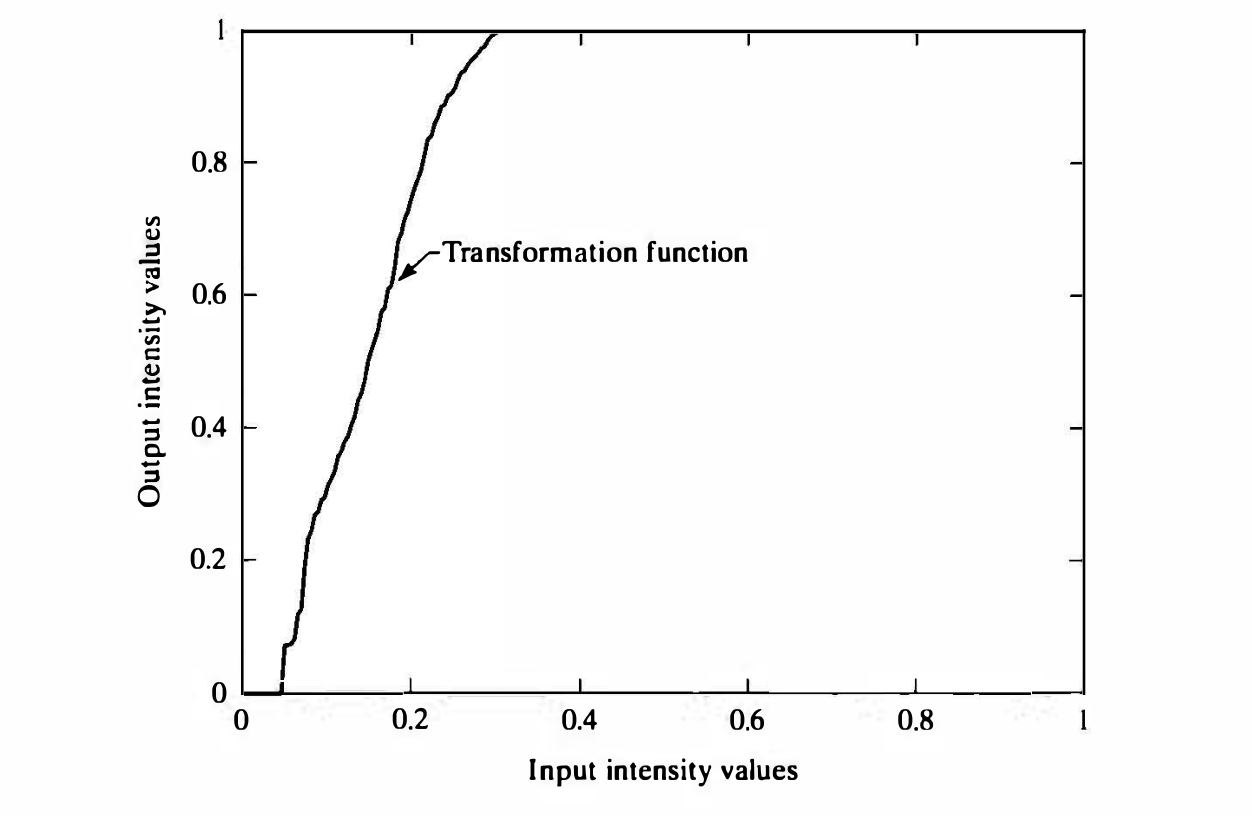
rõ ràng trong biểu đồ của hình ảnh này, được hiển thị trong Hình 3.8(d). Sự gia tăng độ tương phản là do sự trải rộng đáng kể của biểu đồ trên toàn bộ thang cường độ. Sự gia tăng cường độ tổng thể là do mức cường độ tuổi trung bình trong biểu đồ của hình ảnh được cân bằng cao hơn (nhẹ hơn) so với ban đầu . Mặc dù phương pháp cân bằng biểu đồ vừa thảo luận không tạo ra biểu đồ phẳng nhưng nó có đặc tính mong muốn là có thể tăng dải động của các mức cường độ trong ảnh.

Như đã lưu ý trước đó, hàm biến đổi được sử dụng trong cân bằng biểu đồ là tổng tích lũy của các giá trị biểu đồ đã chuẩn hóa. Chúng ta có thể sử dụng hàm cumsum để thu được hàm biến đổi như sau:

>> hnorm = imhist(f)./numel(f); % Biểu đồ chuẩn hóa. >> cdf = cumsum(hnorm) ; % GDF.

Một biểu đồ cdf, được hiển thị trong Hình 3.9, được tạo ra bằng cách sử dụng các lệnh sau:

**Chương 3** • Chuyển đổi cường độ và lọc không gian



Văn bản trong phần nội dung của biểu đồ được chèn bằng các lệnh TextBox và Arrow từ menu Insert trong cửa sổ hình MATLAB chứa biểu đồ. Bạn có thể sử dụng chú thích hàm để viết mã chèn các mục như hộp văn bản và mũi tên trên biểu đồ, nhưng menu Chèn dễ sử dụng hơn đáng kể.

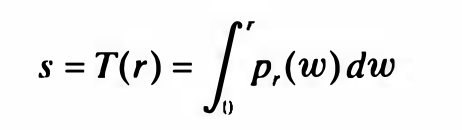
Bạn có thể thấy bằng cách nhìn vào biểu đồ trong Hình 3.8 rằng hàm biến đổi trong Hình 3.9 ánh xạ một phạm vi hẹp các mức cường độ ở đầu dưới của thang cường độ đầu vào tới phạm vi cường độ đầy đủ trong hình ảnh đầu ra. Sự cải thiện về hình ảnh độ tương phản được thể hiện rõ bằng cách so sánh hình ảnh đầu vào và đầu ra trong Hình 3.8.

3.3.3 So khớp biểu đồ (Thông số kỹ thuật)

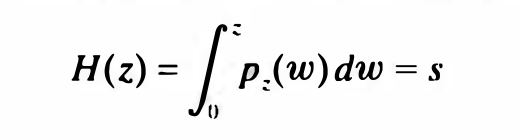
Cân bằng biểu đồ tạo ra một hàm biến đổi có tính thích ứng, theo nghĩa là nó dựa trên biểu đồ của một hình ảnh nhất định. Tuy nhiên, khi hàm biến đổi của một ảnh đã được tính toán, nó sẽ không thay đổi.

trừ khi biểu đồ của hình ảnh thay đổi. Như đã lưu ý trong phần trước, việc cân bằng biểu đồ đạt được sự nâng cao bằng cách trải rộng các mức của hình ảnh đầu vào trên một phạm vi rộng hơn của thang cường độ. Trong phần này, chúng tôi chỉ ra rằng điều này không phải lúc nào cũng dẫn đến kết quả thành công. Đặc biệt, nó rất hữu ích trong một số ứng dụng khi có thể chỉ định hình dạng của biểu đồ mà chúng ta mong muốn hình ảnh được xử lý có. Phương pháp được sử dụng để tạo ra một hình ảnh có biểu đồ cụ thể được gọi là so khớp biểu đồ hoặc đặc tả biểu đồ.

Về nguyên tắc, phương pháp này rất đơn giản. Hãy xem xét các mức liên tục trong giây lát được chuẩn hóa thành khoảng [O, 1] và đặt r và z biểu thị mức cường độ của hình ảnh đầu vào và đầu ra. Các mức đầu vào có hàm mật độ xác suất p,(r) và các mức đầu ra có hàm mật độ xác suất xác định p\_(z). Chúng ta biết từ cuộc thảo luận ở phần trước rằng anh ta biến đổi



dẫn đến mức cường độ, s, với hàm mật độ xác suất thống nhất:



Chúng ta có thể tìm T(r) từ hình ảnh đầu vào (đây là phép biến đổi cân bằng biểu đồ đã thảo luận trong phần trước), do đó, chúng ta có thể sử dụng phương trình trước để tìm các mức biến đổi z có mật độ là p đã chỉ định,( z) với điều kiện là chúng ta có thể tìm được H-1. Khi làm việc với các biến rời rạc, chúng ta có thể đảm bảo rằng nghịch đảo của H tồn tại nếu p(zk ) là togram của nó hợp lệ (tức là nó có diện tích đơn vị và tất cả các giá trị của nó đều không âm) và không có thành phần nào của nó bằng 0 [tức là không có ngăn nào của p(zk ) trống]. Giống như trong việc cân bằng biểu đồ, việc triển khai riêng biệt phương pháp trước đó chỉ mang lại kết quả gần đúng với biểu đồ đã chỉ định.

Hộp công cụ thực hiện khớp biểu đồ bằng cú pháp sau trong lịch sử:

trong đó f là hình ảnh đầu vào, hspec là biểu đồ được chỉ định (một vectơ hàng có giá trị được chỉ định) và g là hình ảnh đầu ra, có biểu đồ gần đúng với biểu đồ được chỉ định, hspec. Vectơ này phải chứa số nguyên tương ứng với các thùng cách đều nhau. Một đặc tính của histeq là biểu đồ của g thường phù hợp hơn với hspec khi độ dài ( hspec ) nhỏ hơn nhiều so với số mức cường độ trong f.

Chương 3 • Biến đổi cường độ và lọc không gian

Hình 3.lO(a) hiển thị hình ảnh f của mặt trăng sao Hỏa, Phobos và Hình 3.lO(b) hiển thị biểu đồ của nó, thu được bằng cách sử dụng imhist ( f ). Hình ảnh bị chi phối bởi các vùng tối, lớn, dẫn đến biểu đồ được đặc trưng bởi sự tập trung lớn các pixel ở phần tối của thang màu xám. Thoạt nhìn, người ta có thể kết luận rằng cân bằng biểu đồ sẽ là một phương pháp tốt để cải thiện hình ảnh này, nhờ đó các chi tiết trong vùng tối trở nên rõ ràng hơn. Tuy nhiên, kết quả trong Hình 3.lO(c), thu được bằng lệnh

>> f1 = histeq(f, 256) ;

cho thấy rằng việc cân bằng biểu đồ trên thực tế đã tạo ra một hình ảnh có vẻ ngoài "bị mờ" - không phải là một kết quả đặc biệt tốt trong trường hợp này. Lý do cho điều này có thể được thấy bằng cách nghiên cứu biểu đồ của ảnh đã được cân bằng, được hiển thị trong Hình 3.10(d). Ở đây, chúng ta thấy rằng các mức cường độ đã được dịch chuyển lên nửa trên của thang màu xám, do đó tạo ra hình ảnh có độ tương phản thấp, mờ nhạt như đề cập ở trên. Nguyên nhân của sự dịch chuyển là do nồng độ lớn các thành phần tối ở hoặc gần 0 trong biểu đồ ban đầu. Hàm biến đổi tích lũy thu được từ biểu đồ này rất dốc, do đó ánh xạ nồng độ lớn các pixel ở đầu thấp của thang màu xám đến đầu cao của thang đo.

Một khả năng để khắc phục tình trạng này là sử dụng kết hợp biểu đồ, với biểu đồ mong muốn có nồng độ thấp hơn các thành phần ở phần cuối của thang màu xám và duy trì hình dạng chung của biểu đồ của ảnh gốc. Chúng tôi lưu ý trong Hình 3.1(b) rằng biểu đồ về cơ bản là lưỡng kim, với một chế độ lớn ở gốc và một chế độ khác, nhỏ hơn, ở đầu cao của thang màu xám. Ví dụ, những loại biểu đồ này có thể được mô hình hóa bằng cách sử dụng các hàm Gaussian đa phương thức. Hàm M sau đây tính toán hàm Gaussian lưỡng kim được chuẩn hóa thành đơn vị diện tích, do đó, nó có thể được sử dụng làm biểu đồ được chỉ định.

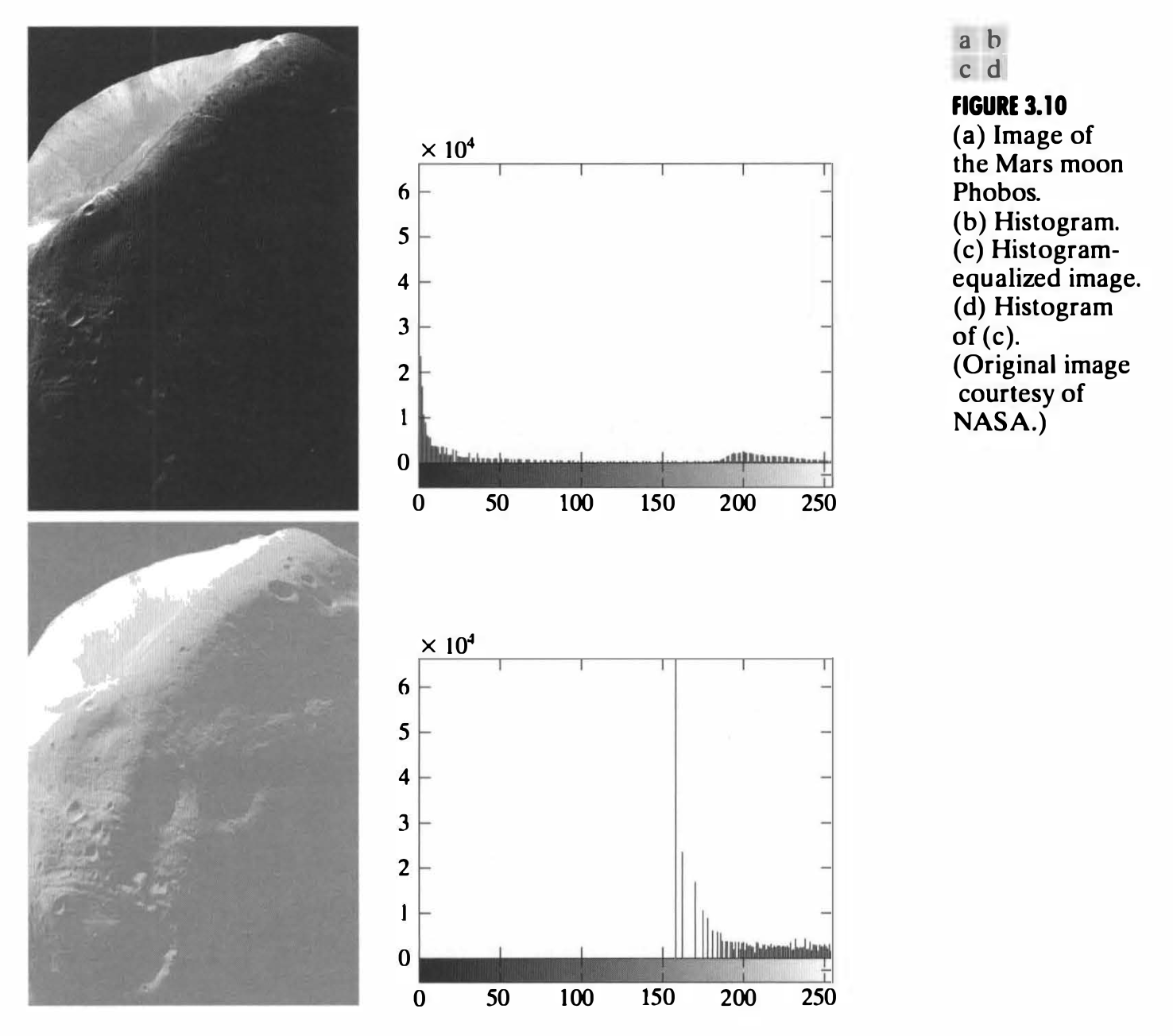
function p = twomodegauss (m1 , sig1 , m2 , sig2 , A1 , A2 , k ) %TWOMODEGAUSS Generates a two-mode Gaussian function.

* %  P = TWOMODEGAUSS(M1 , SIG1 , M2, SIG2, A1 , A2, K) generates a
* %  two-mode, Gaussian-like function in the interval (O, 1 ] . P is a
* %  256-element vector normalized so that SUM(P) = 1. The mean and
* %  standard deviation of the modes are (M1 , SIG1 ) and (M2, SIG2) ,
* %  respectively. A1 and A2 are the amplitude values of the two
* %  modes . Since the output is normalized , only the relative values
* %  of A1 and A2 are important. K is an offset value that raises the
* %  "floor" of the function. A good set of values to try is M1 = % 0.15,SIG1=0.05,M2=0.75,SIG2=0.05,A1= 1,A2= 0.07, % and K = 0.002. sig1); sig2);

c1=A1\* (1I((2\* pi)A 0.5)\* k1 = 2 \* (sig1 A 2)i  
c2 = A2 \* (1 I ((2 \* pi) A 0.5) \* k2 2\* (sig2A 2)i

z = linspace(O, 1, 256)i

3.3 . Xử lý biểu đồ và vẽ đồ thị hàm - 105



p = k + c1 \* exp(-((z - m1) .' 2) ./ k1) + ... p

c2/\* exp(-((z - m2) p . sum(p(:)); / 2) .I k2);

Hàm tương tác sau đây chấp nhận đầu vào từ bàn phím và vẽ đồ thị hàm Gaussian kết quả. Tham khảo Phần 2.10.6 để biết giải thích về chức năng đầu vào. Lưu ý cách đặt giới hạn của các ô.

hàm p = lịch sử thủ công

%MANUALHIST Tạo biểu đồ hai chế độ một cách tương tác.

% P = MANUALHIST tạo biểu đồ hai chế độ bằng hàm

% TWOMODEGAUSS(m1 , sig1 , m2, sig2, A1 , A2, k) . m1 và m2 là

% nghĩa là của hai chế độ và phải nằm trong khoảng [0,1]. SIG1 và

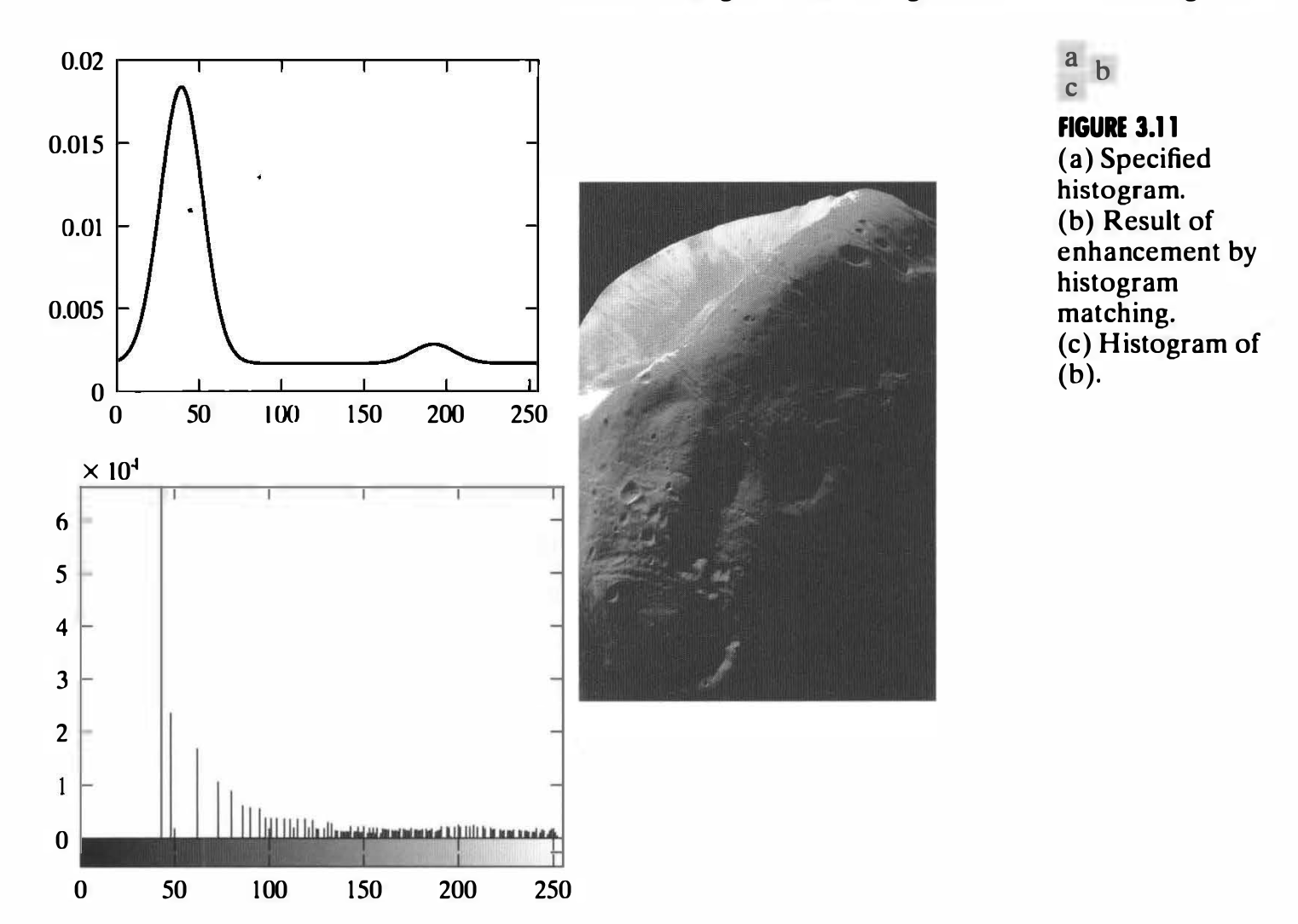
% SIG2 là độ lệch chuẩn của hai chế độ. A1 và A2 là

% giá trị biên độ và k là giá trị bù làm tăng sàn

Chương 3 • Biến đổi cường độ và lọc không gian:

Bởi vì vấn đề cân bằng biểu đồ trong ví dụ này chủ yếu là do sự tập trung lớn các pixel trong ảnh gốc với các mức gần bằng 0, nên cách tiếp cận hợp lý là sửa đổi biểu đồ của hình ảnh đó để nó không có đặc tính này. Hình 3.ll(a) hiển thị đồ thị của hàm (thu được từ chương trình manualhist) giữ nguyên hình dạng chung của biểu đồ ban đầu nhưng có sự chuyển tiếp mượt mà hơn các mức trong vùng..tối của thang cường độ. Đầu ra của chương trình, p, bao gồm 256 điểm cách đều nhau từ hàm này và là biểu đồ được chỉ định mong muốn. Một hình ảnh có biểu đồ được chỉ định đã được tạo bằng lệnh

3.3 • Xử lý biểu đồ và vẽ đồ thị hàm 107



Hình 3.11(b) thể hiện kết quả. Sự cải thiện so với kết quả cân bằng biểu đồ trong Hình.3.1O(c) là hiển nhiên. Lưu ý rằng biểu đồ được chỉ định thể hiện sự thay đổi khá khiêm tốn so với biểu đồ ban đầu. Đây là tất cả những gì cần thiết để đạt được sự cải thiện đáng kể về khả năng cường hóa. Biểu đồ của Hình 3. 1 1 (b) được thể hiện trong Hình 3.1 l (c). Đặc điểm nổi bật nhất của biểu đồ này là cách phần dưới của nó được di chuyển đến gần vùng sáng hơn của thang màu xám và do đó gần với hình dạng được chỉ định hơn. Tuy nhiên, lưu ý rằng sự dịch chuyển sang phải không quá lớn bằng sự dịch chuyển trong biểu đồ trong Hình 3.lO(d), tương ứng với hình ảnh được tăng cường kém của Hình 3.lO(c).

3.3.4 Chức năng thích ứng

Chức năng hộp công cụ này thực hiện cái gọi là cân bằng biểu đồ thích ứng giới hạn độ tương phản (CLAHE). Không giống như các phương pháp được thảo luận ở hai phần trước, hoạt động trên toàn bộ hình ảnh, phương pháp này bao gồm việc xử lý

các vùng nhỏ của hình ảnh (được gọi là các ô) sử dụng thông số biểu đồ cho từng ô riêng lẻ. Các ô lân cận sau đó được kết hợp bằng cách sử dụng phép nội suy song tuyến tính để loại bỏ các ranh giới được tạo ra một cách giả tạo. Sự tương phản, đặc biệt trong

Chương 3 • Biến đổi cường độ và lọc không gian

các vùng có cường độ đồng nhất, có thể được hạn chế để tránh khuếch đại tiếng ồn. Các

VÍ DỤ 3.7:

Sử dụng chức năng

Adapthisteq.

cú pháp cho Adapthisteq là

g = Adapthisteq(f, param1 , val1 , param2, val2, . . . )

trong đó f là ảnh đầu vào, g là ảnh đầu ra và các cặp param/val được liệt kê trong Bảng 3.2.

•Hình 3.12(a)cũng giống như Hình 3.lO(a) và Hình 3.12(b)là kết quả của việc sử dụng tất cả các cài đặt mặc định trong hàm Adapthisteq:

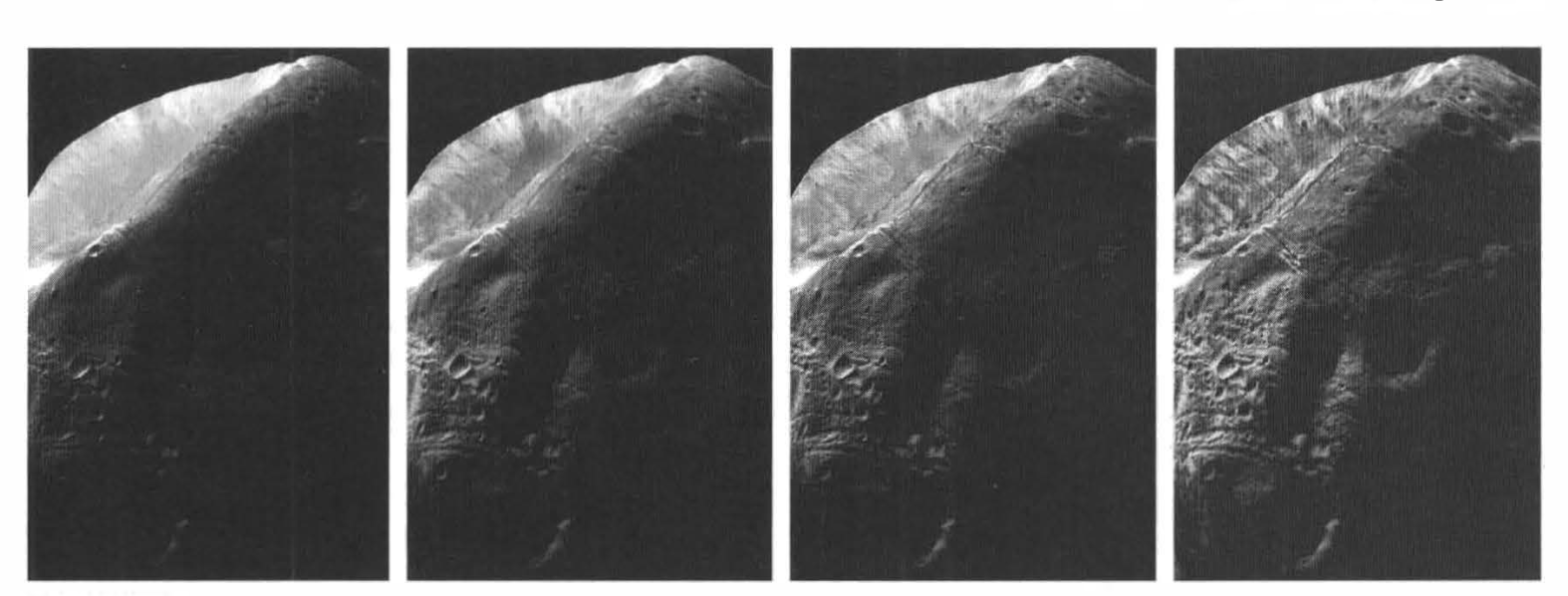
>> g1 = Adapthisteq(f);

Mặc dù kết quả này cho thấy độ chi tiết tăng lên một chút nhưng những phần đáng kể của hình ảnh vẫn nằm trong bóng tối. Hình 3.12(c) cho thấy kết quả của việc tăng kích thước của các ô lên [25 25]:

» g2 = Adapthisteq(f, 'NumTiles', [25 25]);

Độ sắc nét tăng nhẹ nhưng không thấy chi tiết mới. Sử dụng com

mệnh lệnh



3.4: ID Spatial Filtering

Như đã đề cập trong Phần 3.1 và minh họa trong Hình.3.1, việc xử lý vùng lân cận bao gồm ( 1 ) chọn điểm trung tâm, (x, y); (2) thực hiện một thao tác chỉ liên quan đến các pixel trong vùng lân cận được xác định trước về (x, y); (3) để kết quả của thao tác đó là "phản hồi" của quy trình tại thời điểm đó; và (4) lặp lại quy trình cho mọi điểm trong ảnh. Quá trình di chuyển điểm trung tâm sẽ tạo ra các vùng lân cận mới, mỗi vùng cho mỗi pixel trong ảnh đầu vào. Hai thuật ngữ chính được sử dụng để xác định hoạt động này là xử lý vùng lân cận và lọc không gian, với thuật ngữ thứ hai phổ biến hơn. Như được giải thích trong phần sau, nếu các tính toán được thực hiện trên các pixel của vùng lân cận là tuyến tính thì thao tác này được gọi là lọc không gian tuyến tính (thuật ngữ tích chập không gian cũng được sử dụng); mặt khác nó được gọi là không gian phi tuyến

3.4.1 Lọc không gian tuyến tính

Khái niệm lọc tuyến tính có nguồn gốc từ việc sử dụng biến đổi Fourier để xử lý tín hiệu trong miền tần số, một chủ đề được thảo luận chi tiết trong Chương 4. Trong chương này, chúng ta quan tâm đến các hoạt động lọc

được thực hiện trực tiếp trên các pixel của hình ảnh. Việc sử dụng thuật ngữ lọc không gian tuyến tính giúp phân biệt loại quy trình này với lọc miền tần số.

Các phép toán tuyến tính quan tâm trong chương này bao gồm nhân từng pixel trong vùng lân cận với một hệ số tương ứng và tính tổng các kết quả để có được phản hồi tại mỗi điểm (x, y). Nếu vùng lân cận có kích thước mXn thì cần có hệ số mn. Các hệ số được sắp xếp dưới dạng ma trận, được gọi là bộ lọc, mặt nạ, mặt nạ lọc, hạt nhân, mẫu hoặc cửa sổ, với ba thuật ngữ đầu tiên là phổ biến nhất. Vì những lý do sẽ sớm trở nên rõ ràng, các thuật ngữ bộ lọc tích chập, mặt nạ tích chập hoặc nhân tích chập cũng được sử dụng.

Hình 3.13 minh họa cơ chế lọc không gian tuyến tính. Quá trình này bao gồm việc di chuyển tâm của mặt nạ lọc, w, từ điểm này sang điểm khác trong ảnh, / . Tại mỗi điểm (x, y), phản hồi của bộ lọc tại điểm đó là tổng tích của các hệ số bộ lọc và các pixel lân cận tương ứng trong vùng được bao trùm bởi mặt nạ bộ lọc. Đối với mặt nạ có kích thước m X n, chúng ta thường giả sử rằng m = 2a + 1 và n = 2b + I trong đó a và b là các số nguyên không âm. Tất cả những điều này nói lên rằng trọng tâm chính của chúng tôi là các mặt nạ có kích thước lẻ, với kích thước có ý nghĩa nhỏ nhất là 3 X 3. Mặc dù chắc chắn đó không phải là một yêu cầu, nhưng làm việc với các mặt nạ có kích thước lẻ sẽ trực quan hơn vì chúng có điểm trung tâm rõ ràng.

Có hai khái niệm liên quan chặt chẽ với nhau phải được hiểu rõ ràng khi thực hiện lọc không gian tuyến tính. Một là sự tương quan; cái còn lại là tích chập. Tương quan là quá trình truyền mặt nạ w bởi mảng ảnh f theo cách mô tả trong hình 3.13. Về mặt cơ học, tích chập là quá trình tương tự, ngoại trừ việc w được quay 1 80° trước khi truyền nó qua .f. Hai khái niệm này được giải thích tốt nhất bằng một số ví dụ.

Hình 3.14(a) cho thấy hàm một chiều,/, và một mặt nạ, w. Gốc của f được coi là điểm ngoài cùng bên trái của nó. Để thực hiện tương quan của hai hàm số, chúng ta di chuyển w sao cho điểm ngoài cùng bên phải của nó trùng với gốc tọa độ, như Hình 3.14(b) cho thấy. Nhận thấy rằng có những điểm giữa hai hàm không trùng nhau. Cách phổ biến nhất để giải quyết vấn đề này là thêm vào f với số lượng Os cần thiết để đảm bảo rằng sẽ luôn có các điểm tương ứng cho toàn bộ hành trình của w qua .f. Tình huống này được minh họa trong hình 3.14(c).

Bây giờ chúng ta đã sẵn sàng để thực hiện mối tương quan. Giá trị tương quan thứ nhất là tổng tích của hai hàm số ở vị trí như hình 3.14(c). Tổng sản phẩm là 0 trong trường hợp này. Tiếp theo, chúng tôi di chuyển sang bên phải một vị trí và lặp lại quy trình [Hình. 3.14(d)]. Tổng sản phẩm lại bằng 0. Sau bốn ca [Hình. 3.14(e)], chúng ta gặp giá trị khác 0 đầu tiên của mối tương quan. đó là (2)(1) = 2. Nếu chúng ta tiến hành theo cách này cho đến khi w di chuyển hoàn toàn qua/ [hình học kết thúc được hiển thị trong Hình 3.14(f)] thì chúng ta sẽ nhận được kết quả trong Hình 3.14(g). Tập hợp các giá trị này là mối tương quan của w và f. Nếu chúng ta đã đệm w, căn chỉnh phần tử ngoài cùng bên phải với phần tử ngoài cùng bên trái của w được đệm và thực hiện tương quan theo cách vừa giải thích. kết quả sẽ khác (xoay 180°), do đó thứ tự của các hàm đóng vai trò quan trọng trong mối quan hệ tương quan.