

Digital Picture Processing

AZRIEL ROSENFELD

Computer Science Center
University of Maryland
College Park, Maryland

AVINASH C. KAK

School of Electrical Engineering
Purdue University
West Lafayette, Indiana

Eastman Kodak Company
Research Library
Kodak Park Division
Rochester, New York 14650

This is a volume in
COMPUTER SCIENCE AND APPLIED MATHEMATICS
A Series of Monographs and Textbooks

Editor: WERNER RHEINBOLDT

A complete list of titles in this series appears at the end of this volume.



ACADEMIC PRESS New York San Francisco London 1976
A Subsidiary of Harcourt Brace Jovanovich, Publishers

COPYRIGHT © 1976, BY ACADEMIC PRESS, INC.
 ALL RIGHTS RESERVED.
 NO PART OF THIS PUBLICATION MAY BE REPRODUCED OR
 TRANSMITTED IN ANY FORM OR BY ANY MEANS, ELECTRONIC
 OR MECHANICAL, INCLUDING PHOTOCOPY, RECORDING, OR ANY
 INFORMATION STORAGE AND RETRIEVAL SYSTEM, WITHOUT
 PERMISSION IN WRITING FROM THE PUBLISHER.

ACADEMIC PRESS, INC.
 111 Fifth Avenue, New York, New York 10003

United Kingdom Edition published by
 ACADEMIC PRESS, INC. (LONDON) LTD.
 24/28 Oval Road, London NW1

Library of Congress Cataloging in Publication Data

Rosenfeld, Azriel, (date)
 Digital picture processing.

(Computer science and applied mathematics series)
 "Supersedes an earlier book . . . Picture processing
 by computer."

Includes bibliographies.
 I. Optical data processing. I. Kak, Avinash C.,
 joint author. II. Title
 TA1630.R67 621.3819'598 75-16874
 ISBN 0-12-597360-8

PRINTED IN THE UNITED STATES OF AMERICA

Contents

<i>Preface</i>	ix
<i>Acknowledgments</i>	xi
 Chapter 1 Introduction	 1
1.1 Picture Processing	1
1.2 Pictures and Their Computer Representation	2
1.3 A Guide to the Literature	6
References	7
 Chapter 2 Mathematical Preliminaries	 9
2.1 Linear Operations on Pictures	9
2.2 Discrete Picture Transforms	19
2.3 Random Variables	27
2.4 Random Fields	35
References	43
 Chapter 3 Visual Perception	 45
3.1 Brightness and Contrast	46
3.2 Acuity and Contour	48
3.3 Color	55
3.4 Pattern and Texture	57
3.5 Shape and Space	59
3.6 Duration and Motion	61
3.7 Detection and Recognition	61
References	63

Chapter 4 Digitization	64
4.1 Sampling Using an Array of Points	65
4.2 Sampling Using Orthonormal Functions	83
4.3 Quantization of Picture Samples	98
4.4 Bibliographical Notes	106
References	107
 Chapter 5 Compression	 108
5.1 Transform Compression	109
5.2 Predictive Compression	127
5.3 More on Compression Techniques	138
5.4 The Rate Distortion Function	140
5.5 Bibliographical Notes	148
References	150
 Chapter 6 Enhancement	 153
6.1 Quality	154
6.2 Gray Scale Modification	167
6.3 Geometric Correction	175
6.4 Sharpening	179
6.5 Smoothing	192
6.6 Bibliographical Notes	200
References	201
 Chapter 7 Restoration	 203
7.1 The <i>A Priori</i> Knowledge Required in Restoration	205
7.2 Inverse Filtering	211
7.3 Least Squares Filtering	216
7.4 Constrained Deconvolution	222
7.5 Recursive Filtering	235
7.6 Bibliographical Notes	249
References	252
 Chapter 8 Segmentation	 256
8.1 Thresholding	258
8.2 Edge Detection	275
8.3 Matching	296
8.4 Tracking	317
8.5 Bibliographical Notes	328
References	330

Chapter 9 Geometry	333
9.1 Adjacency and Connectedness	335
9.2 Size and Distance	349
9.3 Arcs and Curves	369
9.4 Directionality and Shape	379
9.5 Bibliographical Notes	400
References	401
 Chapter 10 Description	 404
10.1 Preprocessing and Normalization	405
10.2 Properties	417
10.3 Relational Structures	431
10.4 Picture Analysis	440
10.5 Bibliographical Notes	447
References	448
 <i>Index</i>	 451

Exercise 9. If $h_1(x, y) = x/2$, $h_2(x, y) = y/2$, and g is

```

1 1 1
1 1 1
1 1 1

```

(surrounded by 0's), construct f , using both alternatives (2a) and (2b). Note that in the second case, f has more than just the two gray levels 0 and 1. ■

Exercise 10. If $h_1(x, y) = (x-y)/\sqrt{2}$, $h_2(x, y) = (x+y)/\sqrt{2}$ (i.e., the transformation is a 45° rotation), and g is

```

1 1 1 1 1 1

```

(surrounded by 0's), construct f . What if g is

```

      1
     1
    1  ?
   1
  1

```

Note that “digital rotation” is not a one-to-one transformation! ■

If h_1 and h_2 are not known, but a distorted picture g of a known pattern, such as a regular grid, is available, it is possible to determine an approximation to the distortion transformation by measuring the positions of grid points in g . For example, suppose that we are given three neighboring grid points that form a small triangle; let their ideal coordinates (in the regular grid) be (r_1, s_1) , (r_2, s_2) , (r_3, s_3) , and let their positions in g be (u_1, v_1) , (u_2, v_2) , (u_3, v_3) , respectively. We can find a linear transformation

$$x' = ax + by + c, \quad y' = dx + ey + f$$

which maps the three points into their distorted positions by solving the six equations

$$\begin{aligned} u_1 &= ar_1 + bs_1 + c, & v_1 &= dr_1 + es_1 + f \\ u_2 &= ar_2 + bs_2 + c, & v_2 &= dr_2 + es_2 + f \\ u_3 &= ar_3 + bs_3 + c, & v_3 &= dr_3 + es_3 + f \end{aligned}$$

for a , b , c , d , e , and f . This transformation can then be used to correct the distortion of the triangular portion of g surrounded by the lines joining the three points. The process can then be repeated for other triples of grid points. In effect, this method constructs a piecewise linear approximation to the unknown distortion. Of course, higher-order approximations can also be used if desired. An example of a distorted picture that has been corrected by piecewise approximation of the distortion derived from the distorted image of a grid is shown in Fig. 11.

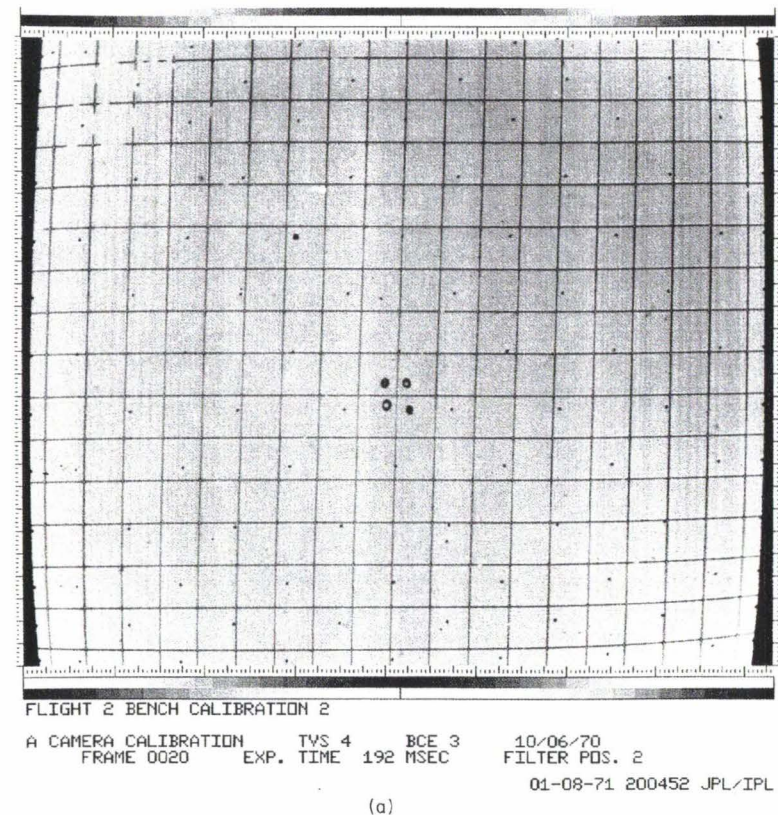


Fig. 11 Geometric correction. (a) Mariner 9 grid target, showing geometrical distortion.

6.4 SHARPENING

As pointed out in Section 6.1.1, picture degradation generally involves blurring, the extent of which is described by the spread functions of the degradation operation. In this section we discuss some simple methods for counteracting blur by “crispning” or “sharpening” the picture. Methods of deblurring based on estimation of the blur will be treated in Chapter 7.

Blurring is an averaging, or integration, operation; this suggests that we may be able to sharpen by performing differentiation operations. Blurring also weakens high spatial frequencies more than low ones; this suggests that

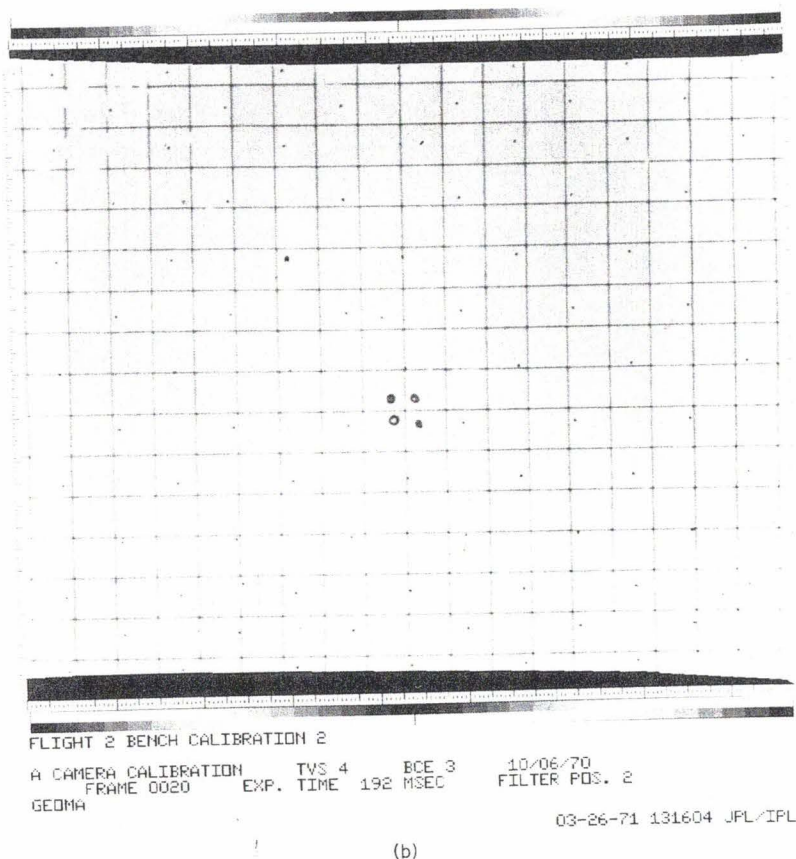


Fig. 11 Geometric Correction. (b) Results of distortion removal. (From D. A. O'Handley and W. B. Green, Recent developments in digital image processing at the Image Processing Laboratory of the Jet Propulsion Laboratory, *Proc. IEEE* 60, 1972, 821-828, Fig. 1.)

pictures can be sharpened by emphasizing their high spatial frequencies. Differentiation operations are discussed in Section 6.4.1, and a particularly useful class of these operations, the "Laplacians," are treated in Section 6.4.2. High-emphasis spatial frequency filtering is treated in Section 6.4.3.

When a picture is noisy as well as blurred, differentiation and high-emphasis filtering cannot be used indiscriminately to sharpen it, since the noise generally involves high rates of change of gray level, and it usually becomes stronger than the picture signal at high frequencies. These methods should be restricted,

if possible, to frequency ranges where the picture is stronger than the noise. Alternatively, one should attempt to remove or reduce the noise before attempting to sharpen the picture. Methods of noise removal and reduction are treated in Section 6.5.

6.4.1 Differentiation

Any partial derivative operator $D = \partial^n / \partial x^k \partial y^{n-k}$ is a linear operator; it follows that any linear combination of D 's is also a linear operator. Any arbitrary combination of D 's is a local operator, since its value for a picture f at a point (x, y) depends only on the values of f in any small neighborhood of (x, y) . This implies that such operators are all shift invariant. We assume in this section that all the derivatives of f exist and are continuous.

It is of particular interest to construct derivative operators that are *isotropic*, i.e., rotation invariant (in the sense that rotating f and then applying the operator gives the same result as applying the operator to f and rotating the output). We want our operators to be isotropic because we want to sharpen blurred features, such as edges and lines, that run in any direction. It can be shown that

- (a) An isotropic linear derivative operator can involve only derivatives of even orders.
- (b) In an arbitrary isotropic derivative operator, derivatives of odd orders can occur only raised to even powers.

We shall now illustrate, by a simple example, how isotropic derivative operators can be constructed. A rotation is defined by the coordinate transformation equations

$$x = x' \cos \theta - y' \sin \theta, \quad y = x' \sin \theta + y' \cos \theta$$

where (x, y) are the unrotated and (x', y') are the rotated coordinates. The first partial derivatives of a picture f in the (x', y') system are given, in terms of its first partials in the (x, y) system, by

$$\frac{\partial f}{\partial x'} = \frac{\partial f}{\partial x} \frac{\partial x}{\partial x'} + \frac{\partial f}{\partial y} \frac{\partial y}{\partial x'} = \frac{\partial f}{\partial x} \cos \theta + \frac{\partial f}{\partial y} \sin \theta,$$

$$\frac{\partial f}{\partial y'} = \frac{\partial f}{\partial x} \frac{\partial x}{\partial y'} + \frac{\partial f}{\partial y} \frac{\partial y}{\partial y'} = -\frac{\partial f}{\partial x} \sin \theta + \frac{\partial f}{\partial y} \cos \theta$$

These partial derivatives themselves are thus not rotation invariant; but the sum of their squares is invariant, since we readily have

$$\left(\frac{\partial f}{\partial x'} \right)^2 + \left(\frac{\partial f}{\partial y'} \right)^2 = \left(\frac{\partial f}{\partial x} \right)^2 + \left(\frac{\partial f}{\partial y} \right)^2$$

Assertion (b) suggests that this sum of squares is of the simplest possible form for an isotropic derivative operator (ignoring degenerate operators that involve no derivatives at all!), since it uses only first-order derivatives raised to the smallest possible even power.

Exercise 11. Prove that the Laplacian operator $(\partial^2/\partial x^2) + (\partial^2/\partial y^2)$ is rotation invariant. (By assertion (a) this has the simplest possible form for a nondegenerate linear isotropic derivative operator, since it uses only derivatives of the smallest possible even order.) ■

As the previous equations show, the partial derivative of a picture f in an arbitrary direction (e.g., $\partial f/\partial x'$) is a linear combination of its partial derivatives in the x - and y -directions. In fact, given the derivatives in any two non-collinear directions, not necessarily perpendicular to each other, we can express the derivative in any other direction as a linear combination of them.

We can find the direction in which the partial derivative of f has maximum value by differentiating $\partial f/\partial x'$ with respect to θ , setting the result equal to zero, and solving for θ . This gives us

$$-\frac{\partial f}{\partial x} \sin \theta + \frac{\partial f}{\partial y} \cos \theta = 0$$

so that the desired direction is

$$\theta_n = \tan^{-1} \left(\frac{\partial f}{\partial y} / \frac{\partial f}{\partial x} \right)$$

Note that $\theta_n + \pi$ is also a solution; in one of these directions, $\partial f/\partial x'$ is a maximum, while in the other direction it is a minimum, since it has equal magnitude but opposite sign, i.e., $\partial f/\partial(-x') = -\partial f/\partial x'$. For this θ_n we have

$$\begin{aligned} \cos \theta_n &= \frac{1}{\sqrt{1 + \tan^2 \theta_n}} = \frac{\partial f}{\partial x} / \sqrt{\left(\frac{\partial f}{\partial x}\right)^2 + \left(\frac{\partial f}{\partial y}\right)^2} \\ \sin \theta_n &= \sqrt{1 - \cos^2 \theta_n} = \frac{\partial f}{\partial y} / \sqrt{\left(\frac{\partial f}{\partial x}\right)^2 + \left(\frac{\partial f}{\partial y}\right)^2} \end{aligned}$$

Thus the maximum value of $\partial f/\partial x'$ is given by

$$\frac{\partial f}{\partial x} \cos \theta_n + \frac{\partial f}{\partial y} \sin \theta_n = + \sqrt{\left(\frac{\partial f}{\partial x}\right)^2 + \left(\frac{\partial f}{\partial y}\right)^2}$$

The vector whose magnitude is $\sqrt{(\partial f/\partial x)^2 + (\partial f/\partial y)^2}$, and whose direction is θ_n (or $\theta_n + \pi$, whichever gives the positive partial derivative) is called the *gradient* of f . As we saw previously the magnitude of the gradient is a rotation-invariant derivative operator. The gradient, and some of its generalizations,

are discussed further in Section 8.2.1, where we deal with edge detection operations.

Exercise 12. Prove that the partial derivative of f is zero in the direction θ_t perpendicular to θ_n . ■

Exercise 13. Prove that the magnitude of the gradient is equal to $\sqrt{(\partial f/\partial \alpha)^2 + (\partial f/\partial \beta)^2}$, where α and β are any two perpendicular directions. ■

For digital pictures, we must use differences rather than derivatives. The first differences in the x - and y -directions are

$$\Delta_x f(i, j) \equiv f(i, j) - f(i-1, j)$$

$$\Delta_y f(i, j) \equiv f(i, j) - f(i, j-1)$$

First differences in other directions can be defined as linear combinations of the x - and y -differences, e.g., as

$$\Delta_\theta f(i, j) = \Delta_x f(i, j) \cos \theta + \Delta_y f(i, j) \sin \theta$$

Using this definition, the maximum directional difference, which we call the *digital gradient*, has magnitude and direction

$$\sqrt{(\Delta_x f)^2 + (\Delta_y f)^2} \quad \text{and} \quad \tan^{-1} (\Delta_y f / \Delta_x f)$$

respectively. (Several other definitions of the digital gradient will be given in Section 8.2.1.)

Exercise 14. What happens to the magnitude of the digital gradient under rotation of the digital picture (see Section 6.3)? ■

The higher difference operators can be defined by repeated first differencing, e.g.,

$$\begin{aligned} \Delta_x^2 f(i, j) &\equiv \Delta_x f(i+1, j) - \Delta_x f(i, j) \\ &= [f(i+1, j) - f(i, j)] - [f(i, j) - f(i-1, j)] \\ &= f(i+1, j) + f(i-1, j) - 2f(i, j) \end{aligned}$$

and similarly

$$\Delta_y^2 f(i, j) = f(i, j+1) + f(i, j-1) - 2f(i, j)$$

We have used these definitions [rather than, e.g., $\Delta_x f(i, j) - \Delta_x f(i-1, j)$] because they are symmetric with respect to (i, j) . We could have also used symmetrical first differences, i.e.,

$$\Delta_x f(i, j) = f(i+1, j) - f(i-1, j)$$

and

$$\Delta_y f(i, j) = f(i, j+1) - f(i, j-1)$$

but we did not do so because they ignore the gray level at (i, j) itself.

6.4.2 The Laplacian

The *Laplacian* is the linear derivative operator

$$\nabla^2 f \equiv \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2}$$

As indicated in Exercise 11, it is rotation invariant.

To see the relevance of the Laplacian to picture sharpening, suppose that the blur in the picture is a result of a diffusion process that satisfies the well-known partial differential equation

$$\partial g / \partial t = k \nabla^2 g$$

where g is a function of x , y , and t (time), and $k > 0$ is a constant. At $t = 0$, $g(x, y, 0)$ is the unblurred picture $f(x, y)$; at some $t = \tau > 0$, we have the observed blurred picture $g(x, y, \tau)$. If we expand $g(x, y, t)$ in a Taylor series around $t = \tau$, we have

$$g(x, y, 0) = g(x, y, \tau) - \tau \frac{\partial g}{\partial t}(x, y, \tau) + \frac{\tau^2}{2} \frac{\partial^2 g}{\partial t^2}(x, y, \tau) - \dots$$

If we ignore the quadratic and higher-order terms, and substitute f for $g(x, y, 0)$ and $k \nabla^2 g$ for $\partial g / \partial t$, this gives us

$$f = g - k\tau \nabla^2 g.$$

Thus, to a first approximation, we can restore the unblurred picture f by subtracting from g a positive multiple of its Laplacian. (Higher-order approximations based on the Taylor series expansion could also be used, if desired.)

Diffusion is not necessarily an appropriate model for picture blur; but the foregoing at least makes it plausible that we can achieve sharpening by a subtractive combination of the picture and its Laplacian. Other rationales for the relevance of the Laplacian to picture sharpening will be given later. It should be mentioned that according to the diffusion model, a point source blurs into a spot with a Gaussian distribution of brightness whose variance is proportional to $k\tau$; we can thus estimate $k\tau$ by fitting a Gaussian to the point spread function.

Several modifications to this Laplacian method have been proposed, with the aim of reducing noise sensitivity. One suggestion is to use, instead of the Laplacian, the second partial derivative in the gradient direction θ_n . A further refinement is to use a linear combination of the second partial derivatives in the two perpendicular directions θ_n and θ_t . This last scheme can be designed to smooth the picture in the direction along an edge at the same time that it deblurs the edge (see also Section 6.5.2).

For a digital picture, the discrete analog of the Laplacian is

$$\begin{aligned} \nabla^2 f(i, j) &\equiv \Delta_x^2 f(i, j) + \Delta_y^2 f(i, j) \\ &= [f(i+1, j) + f(i-1, j) + f(i, j+1) + f(i, j-1)] - 4f(i, j) \end{aligned}$$

Note that this is proportional, by the factor $-\frac{1}{3}$, to

$$f(i, j) - \frac{1}{3}[f(i+1, j) + f(i-1, j) + f(i, j) + f(i, j+1) + f(i, j-1)]$$

which is the difference between the gray level $f(i, j)$ and the average gray level in a neighborhood of (i, j) , where the neighborhood consists of (i, j) and its four horizontal and vertical neighbors. Thus we see that the digital Laplacian of a picture f is obtained, up to a constant factor, by subtracting a blurred (i.e., averaged) version of f from f itself.

Alternative digital "Laplacians" can be defined by using different neighborhoods (e.g., the 3×3 neighborhood consisting of (i, j) and its 8 horizontal, vertical, and diagonal neighbors), or by using a weighted average over the neighborhood. One could also combine the original and blurred pictures nonlinearly, e.g., divide by the blurred picture instead of subtracting it. The operation of subtracting a blurred version of a picture from the original picture (e.g., adding a blurred negative to a sharp positive) is a well-known one in photography, where it is called "unsharp masking."

The effect of the digital Laplacian on a picture can be understood by considering a simple one-dimensional example. Suppose that the $1 \times n$ digital "picture" f contains the sequence of gray levels

$$\begin{aligned} \dots, 0, 0, 0, 1, 2, 3, 4, 5, 5, 5, 5, 5, \\ 5, 6, 6, 6, 6, 6, 6, 3, 3, 3, 3, \dots \end{aligned}$$

It is easily verified that $\Delta_x^2 f$ at these points has the values

$$\begin{aligned} \dots, 0, 0, 1, 0, 0, 0, 0, -1, 0, 0, 0, 0, \\ 1, -1, 0, 0, 0, 0, -3, 3, 0, 0, 0, \dots \end{aligned}$$

Note that this is zero on flat intervals or ramps, as is appropriate for a second difference operator; and it is nonzero at the top and bottom of a ramp, and just on each side of a step. (The use of the Laplacian as an edge detector will be discussed in Section 8.2.) If we subtract $\Delta_x^2 f$ from f , point by point, we obtain

$$\begin{aligned} \dots, 0, 0, -1, 1, 2, 3, 4, 6, 5, 5, 5, 5, \\ 4, 7, 6, 6, 6, 6, 9, 0, 3, 3, 3, \dots \end{aligned}$$

We have now created gray level "undershoots" at the bottom of the ramp and on the low sides of the edges, and "overshoots" at the ramp top and on

the high sides of the edges. This has the effect of increasing the average ramp steepness, and of increasing the contrast just at the edges. Thus we have confirmed, in this case, that subtracting the Laplacian from the picture should have a deblurring effect.

In two dimensions, $f(i, j) - \nabla^2 f(i, j)$ is

$$5f(i, j) - [f(i+1, j) + f(i-1, j) + f(i, j+1) + f(i, j-1)]$$

Readily, we have $f - \nabla^2 f = f$ when (i, j) is in the middle of a ramp or flat region. If (i, j) is just at the bottom of a ramp, or on the low side of an edge, so that some of the neighbors' gray levels are higher than $f(i, j)$, while none are lower, we have $(f - \nabla^2 f) < f$ at (i, j) , which produces an undershoot. Similarly, at the top of a ramp or on the high side of an edge, none of the neighbors' gray levels are higher than $f(i, j)$, and some are lower, so that $(f - \nabla^2 f) > f$ at (i, j) , producing an overshoot. A simple two-dimensional example illustrating these phenomena is shown in Fig. 12; some real cases are shown in Fig. 13.

We can obtain further insight into why subtracting a digital picture's Laplacian from the picture should have a deblurring effect by considering one last example. Suppose that the original picture f was blurred by adding to the

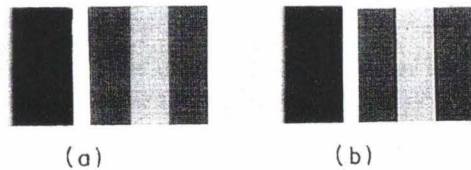


Fig. 12 Results of subtracting the Laplacian from the original picture in some simple cases. (a) Original pictures (ramp and bar). (b) Pictures minus Laplacians.

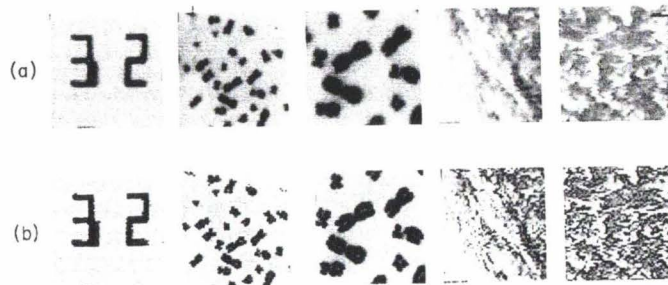


Fig. 13 Results of subtracting the Laplacian from the original picture in some real cases. (a) Original pictures. (b) Pictures minus Laplacians (negative values have been set to 0, and values beyond the end of the gray scale have been set to the highest gray level).

gray level of each point a small fraction ϵ of the sum of its four neighbors' levels, yielding the blurred picture g in which

$$g(i, j) = f(i, j) + \epsilon s(i, j)$$

where $s(i, j) = f(i+1, j) + f(i-1, j) + f(i, j+1) + f(i, j-1)$. Let us subtract from g a multiple (by θ , say) of its Laplacian, obtaining

$$(1 + 4\theta)g(i, j) - \theta[g(i+1, j) + g(i-1, j) + g(i, j+1) + g(i, j-1)]$$

If we substitute for the g 's in this expression, we obtain

$$(1 + 4\theta)f(i, j) + (1 + 4\theta)\epsilon s(i, j)$$

$$- \theta s(i, j) - \theta \epsilon [s(i+1, j) + s(i-1, j) + s(i, j+1) + s(i, j-1)]$$

Let us choose θ so that the coefficient of $s(i, j)$, which is $(1 + 4\theta)\epsilon - \theta$, vanishes; this requires $\theta = \epsilon/(1 - 4\epsilon)$. If ϵ is sufficiently small, we can ignore terms of order ϵ^2 ; in particular, we can ignore the last term, which is a multiple of $\theta\epsilon \doteq \epsilon^2$. Hence

$$g(i, j) - \frac{\epsilon \nabla^2 g(i, j)}{1 - 4\epsilon} \doteq \left(1 + \frac{4\epsilon}{1 - 4\epsilon}\right)f(i, j)$$

Thus by subtracting a suitable multiple of g 's Laplacian from g , we have approximately restored the unblurred f .

Exercise 15. In optical microscopy, let f_k be the cross section of the specimen at depth k . When we focus on that cross section, the image g_k that we obtain is degraded by the presence of blurred images of neighboring cross sections, say $g_k \doteq f_k + \epsilon(\bar{f}_{k+1} + \bar{f}_{k-1})$, where the overbars denote local averaging. Show how f_k can be restored, to a first approximation, by subtracting a suitable multiple of $(\bar{g}_{k+1} + \bar{g}_{k-1})$ from g_k . (On this method see M. Weinstein and K. R. Castleman, *Reconstructing 3-D specimens from 2-D section images*, in "Quantitative Imagery in the Biomedical Sciences," pp. 131–137. R. E. Herron (ed.), S.P.I.E., Redondo Beach, California, 1972.) ■

6.4.3 High-Emphasis Filtering

The effects of derivative operators in sharpening a picture can also be interpreted from a spatial frequency standpoint. Since the derivative of $\sin nx$ is $n \cos nx$, we see that the higher the spatial frequency of a sinusoidal pattern, the higher the amplitude of its derivative is. Thus differentiation strengthens high spatial frequencies more than it does low ones. (Conversely, the integral of $\cos nx$ is $(1/n) \sin nx$, so that integration weakens high frequencies more than it does low ones.)

The effect of subtracting a Laplacian from a picture can be explained along similar lines. As we saw in Section 6.4.2,

$$f(i, j) - \nabla^2 f(i, j) = f(i, j) + 5\{f(i, j) - \frac{1}{5}[f(i+1, j) + f(i-1, j) + f(i, j+1) + f(i, j-1)]\} + f(i, j)$$

Here the second term is proportional to the difference between the original picture f and a blurred version of f , call it $\bar{f}$. Now in $\bar{f}$, high spatial frequencies have been weakened more than low ones. Hence when we subtract $\bar{f}$ from f , since the Fourier transform is a linear operator, the low frequencies in f are more or less cancelled out, while the high frequencies remain relatively intact. Thus, when we add a multiple of $f - \bar{f}$ to f , we are boosting the high frequencies, while leaving the low ones relatively unaffected.

These qualitative remarks can help to justify the use of Laplacians and other derivative operators to sharpen pictures. These methods should, however, be considered only in the absence of knowledge about the blur that is to be corrected. When such knowledge is available, restoration techniques such as inverse filtering (Section 7.2) should be used instead. In such techniques, the degree of emphasis on the high frequencies, as a function of frequency, is chosen so as to just compensate for the weakening caused by the blurring. If noise is present in addition to the blur, however, frequencies at which the noise is strong should not be emphasized. Typically, a high-emphasis filtering operation designed for blur compensation has a transfer function which increases with increasing spatial frequency, up to the point where noise would begin to dominate; it then drops off, more or less abruptly, to zero. The form of the dropoff can be designed to simulate some familiar type of transfer function; this helps avoid artifacts in the sharpened picture. A good example of the type of enhancement that can be achieved by this type of filtering is shown in Fig. 14.

By the Convolution theorem (Section 2.1.4), convolving a picture g with some function h gives the same result as multiplying the Fourier transform G of g by the Fourier transform H of h , and then taking the inverse transform. We also saw in Section 2.1.2 that any shift-invariant linear operation is a convolution operation. Analogous results hold for discrete convolutions, using the discrete Fourier transform (Section 2.2.1). Now the digital Laplacian ∇^2 is a shift-invariant linear operation; in fact, applying it to a picture g is the same as convolving g with the array h_L defined by

$$\begin{array}{ccc} & 1 & \\ 1 & -4 & 1 \\ & 1 & \end{array}$$

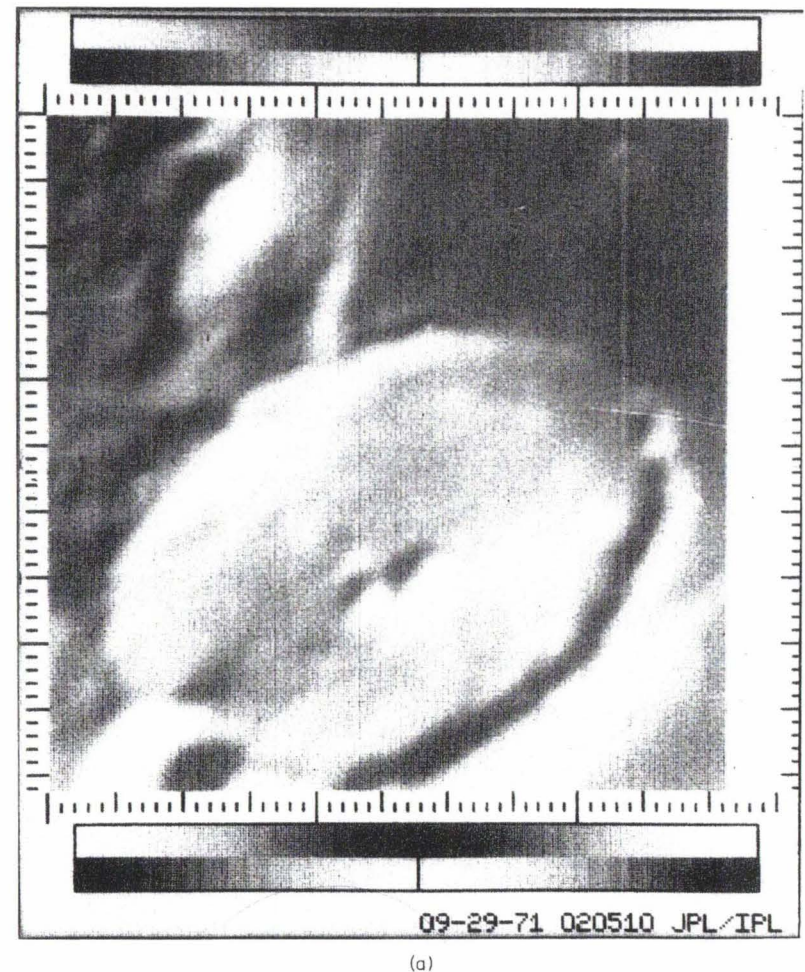


Fig. 14 High-emphasis spatial frequency filtering. (a) The lunar crater Gassendi, blurred by atmospheric turbulence.

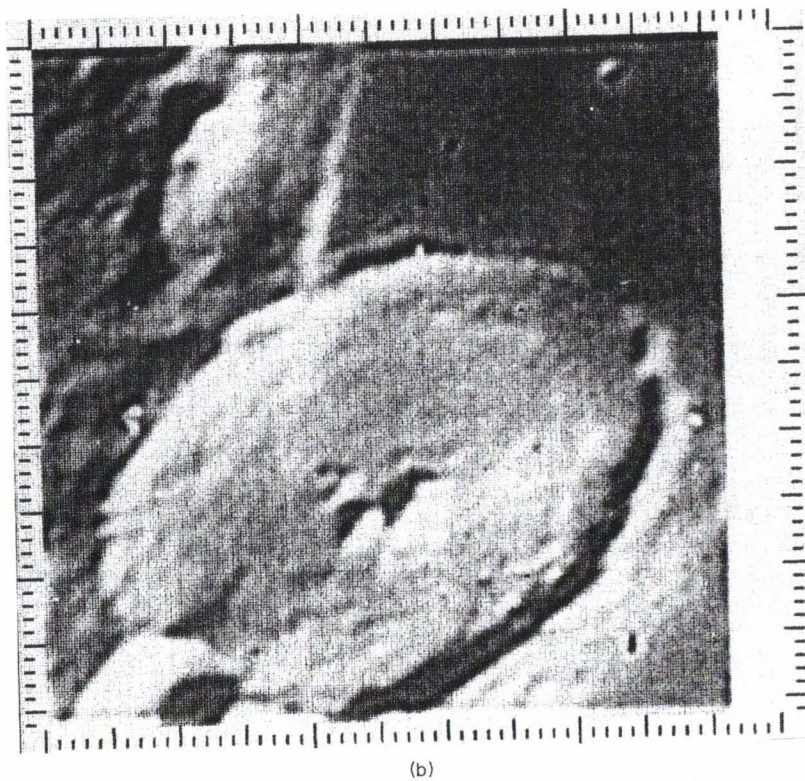


Fig. 14 High-emphasis spatial frequency filtering. (b) Results of high-emphasis filtering. (From D. A. O'Handley and W. B. Green, Recent developments in digital image processing at the Image Processing Laboratory at the Jet Propulsion Laboratory, *Proc. IEEE* 60, 1972, 821-828, Fig. 6.)

in which all points not shown have value 0. Similarly, computing $g - \nabla^2 g$ is the same as convolving g with h_D , defined by

$$\begin{array}{ccc} & -1 & \\ -1 & 5 & -1 \\ & -1 & \end{array}$$

These space-domain convolution operations can also be performed in the

frequency domain, by multiplying the discrete Fourier transform H_L (or H_D) by G and then inverse transforming.

Exercise 16. Compute the discrete Fourier transform of the array h_D (which you may assume to be $N \times N$). ■

Conversely, given any frequency-domain high-emphasis filter H , we can apply it to a picture g in two ways: multiply g 's Fourier transform G by H and inverse transform; or convolve g with the inverse Fourier transform h of H . The h 's obtained in this way will generally have both positive and negative values, but $|h|$ will drop off rapidly from a central peak; thus a good approximation to the desired filtering can be obtained using only a small piece of h surrounding the peak. In the digital case, we can still obtain reasonable results if we take this piece to be as small as 3×3 ; the resulting truncated h generally resembles the $g - \nabla^2 g$ operator, having a positive peak symmetrically surrounded by smaller negative values, whose magnitudes add up to about 1 less than the value at the peak.

Exercise 17. Compute the h that corresponds to the one-dimensional high-emphasis filter H for which $H(n) = |n| + 1$, $-3 \leq n \leq 4$ (i.e., $N = 8$). ■

It is sometimes advantageous to combine spatial frequency filtering with a nonlinear transformation of the gray scale. As an important example of this, suppose that we want to remove shading effects, due to uneven illumination, from a picture, without degrading the detail in the picture. Now the illumination incident on an object usually varies more slowly than does the object's reflectance. Thus uneven illumination has more significant effects at low than at high spatial frequencies, and its effects can be reduced by high-emphasis filtering. In this connection, however, it is important that, in the picture being filtered, brightness be represented on a logarithmic scale. (An example of such a scale is photographic density, which is proportional to the logarithm of transmittance or reflectance of the photograph, and is therefore usually a logarithmic function of the exposure.) Indeed, suppose that the illumination and reflectance are $i(x, y)$ and $r(x, y)$, respectively; then the resultant object brightness is given by $b(x, y) = i(x, y)r(x, y)$. It follows that the logarithm of brightness is

$$\log b(x, y) = \log i(x, y) + \log r(x, y)$$

in which i and r enter additively, rather than multiplicatively. Since the Fourier transform is a linear operation, it is now possible to deemphasize selectively the effects of $i(x, y)$ by high-emphasis filtering. If desired, the results can be redisplayed on a linear brightness scale by performing a pointwise exponentiation. An example of a picture which has been processed in this way is shown in Fig. 15.

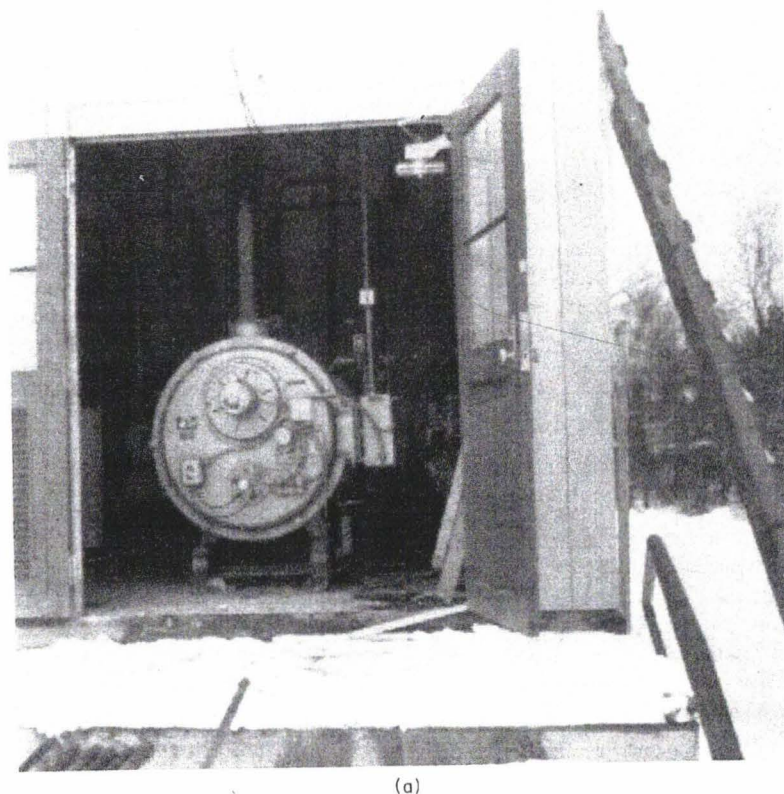


Fig. 15 High-emphasis filtering using a logarithmic transformation. (a) Original picture.

6.5 SMOOTHING

As we saw in Section 6.1.4, many types of noise can be present in a picture. In this section we describe some simple methods of noise removal, or more generally, of making a picture "smoother." Optimal estimation of a picture that has been degraded by noise will be treated in Section 7.5.

The basic difficulty with noise removal and smoothing techniques is that, if applied indiscriminately, they tend to blur the picture, which is usually objectionable. In particular, one usually wants to avoid blurring sharp edges

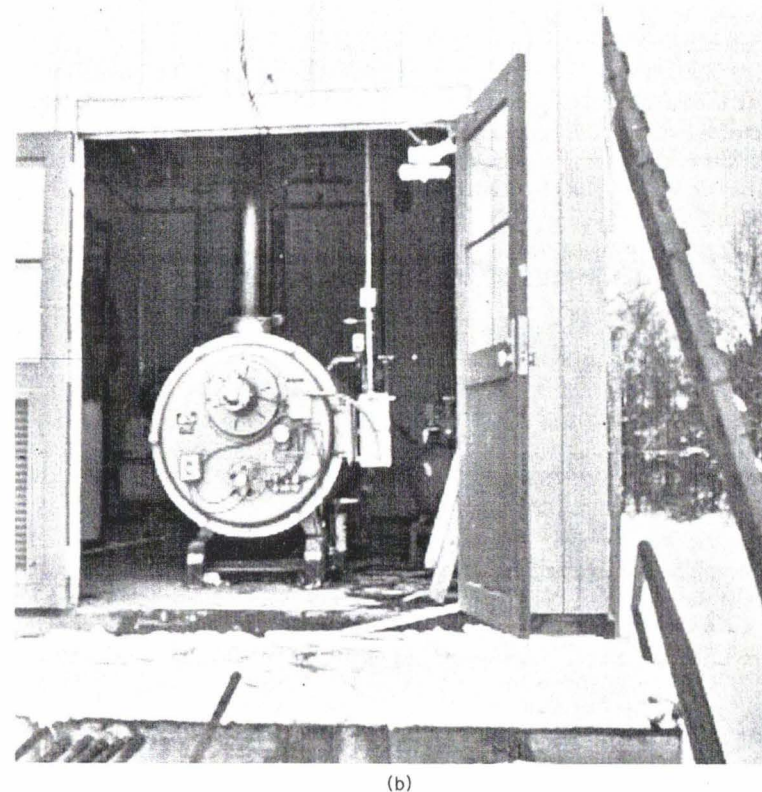


Fig. 15 High-emphasis filtering using a logarithmic transformation. (b) Results of filtering. (From T. G. Stockham, Jr., Image processing in the context of a visual model, *Proc. IEEE* 60, 1972, 828-842, Fig. 11.)

or lines that occur in the picture. Our main concern in this section will be with methods which permit smoothing without introducing undesirable blurring.

A general approach to picture smoothing would be to define a cost function φ for evaluating the various possible smoothings $\hat{f}$ of a given noisy picture g . This φ should depend both on the irregularity of $\hat{f}$ and on the discrepancy between $\hat{f}$ and g . For example, φ could be a weighted integral over the picture of $(\partial\hat{f}/\partial x)^2 + (\partial\hat{f}/\partial y)^2 + (\hat{f} - g)^2$; the first two terms give the mean magnitude of the gradient of $\hat{f}$, while the third term is the mean squared difference between

$\hat{f}$ and g . The weighting function in the integral should be defined so that φ is sensitive to noisy fluctuations in $\hat{f}$, but not to high values of $\hat{f}$'s gradient that are due to the presence of sharp long edges or lines, which are presumably not noise. Given φ , one can attempt to find a smoothed f that minimizes it; if we used a discrete version of φ , defined as a sum over the points of the digital pictures $\hat{f}$ and g , this minimization can be done, in principle, using mathematical programming techniques. In practice, however, since the number of variables in this global optimization problem is very large, it may be too difficult to find the optimum $\hat{f}$ in this way unless the picture g is small. The nonoptimal techniques described in this section involve only local processing of g , and should thus be much simpler computationally.

6.5.1 Noise Removal

If the noise occurs in known positions in the picture, or if we are able to distinguish the noise from the rest of the picture, it becomes relatively easy to remove the noise without bad effects on other parts of the picture, since we can operate only on the noise, leaving the rest of the picture intact.

As a simple example, suppose that a periodic line pattern v has been added to the picture f , as in Fig. 16a. We can remove the lines, without affecting the rest of the picture, by changing the gray level of each line point to the average of the levels of neighboring picture points. The results of line removal by this type of "interpolation" are shown in Fig. 16b. Alternatively, we can remove

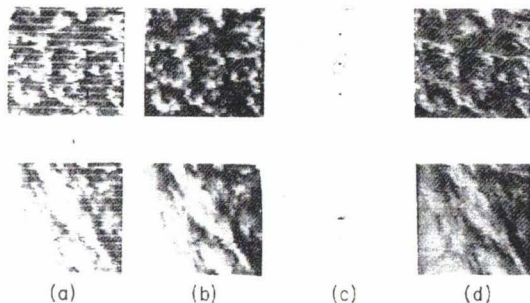


Fig. 16 Suppression of periodic noise by spatial frequency filtering. (a) Original pictures. (b) Results of removing the lines by interpolation (each point on a line is replaced by the average of the points above and below). (c) Power spectra; note high-value points corresponding to spectrum of lines. (d) Results of removing small neighborhoods of these high-value points and reconstructing the pictures.

the lines by operating in the Fourier domain. If we take the Fourier transform of the noisy picture $f+v$, we obtain the sum $F+N$ of the transforms of f and v (see Fig. 16c). Now the Fourier transform N of the line pattern v has all its energy concentrated at a set of small spots along a line perpendicular to the direction of the lines. If we suppress these spots from the transform (zero them out, or remove them by interpolation), and then take the inverse transform, we obtain a smoothed picture in which v has been deleted, while f is relatively unaffected, as shown in Fig. 16d.

Exercise 18. Describe how to remove a line pattern which has been combined multiplicatively, rather than additively, with a picture (i.e., fv rather than $f+v$). ■

It should be pointed out that interpolation to replace the removed noise is easy when the noise is fine (isolated points, thin lines, etc.), so that each noise point has nonnoise neighboring points. When the noise is coarse (large artifacts, wide bars, etc.) it becomes much harder to replace it inconspicuously.

Another case in which noise removal is relatively easy is when the noise consists of isolated points that contrast with their neighbors (e.g., "salt-and-pepper" noise). Here we can attempt to detect noise points by comparing each point's gray level z with the levels z_i of its neighbors. If z is substantially larger (or smaller) than all, or nearly all, of the z_i , we can classify it as a noise point, and remove it by interpolation, i.e., replace z by the average of the z_i . An example of noise removal by this method is shown in Fig. 17.

The method just described has several parameters that can be adjusted to suit the characteristics of the noise that is to be detected. These parameters include the neighborhood size, the threshold amount Δ by which z and the z_i must differ ("substantially"), and the number k of z_i from which z must differ ("nearly all"). The parameter Δ might be defined as some multiple of the estimated standard deviation σ of the noise; as pointed out in Section

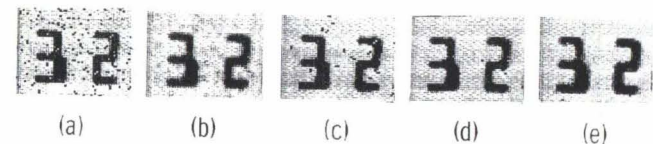


Fig. 17 Suppression of "salt-and-pepper" noise by selective local averaging. (a) Noisy picture. (b) Results of blurring (a)—each point replaced by the average of its eight neighbors. (c)–(e) Results of doing the replacement only for points that differ by at least three gray levels from k of their neighbors, for $k = 8, 7, 6$.

6.1.4, we can estimate σ by measuring the standard deviation of gray level over a region that we believe to be constant in the nonnoisy picture. (We assume here that the noise has zero mean.) The choice of parameter k is important in discriminating between noise points and points that lie on edges or lines; the latter type of point will differ from some of its neighbors, but should not differ from “nearly all” of them.

A simplified version of this method, which is often proposed, is to compare z to the average of the z_i (i.e., to examine the value of the digital Laplacian at the given point), rather than to the z_i individually. This approach, however, would be less able to distinguish isolated points from points on edges or lines, since the value of the Laplacian at a low-contrast isolated point can be the same as its value on a medium-contrast line or a high-contrast edge, as we will see in Section 8.2.1. Simplification of the method is possible, however, when the given picture is two-valued (“black and white”). Here we can detect noise points by counting the number of their neighbors from which they differ. A black point that has too many (“nearly all”) white neighbors can be changed from black to white, and vice versa. (Replacement by averaging would not be appropriate here, since we want the cleaned picture to remain two-valued.)

As just described the method makes a forced-choice decision as to whether or not the given point is a noise point. If we decide that the point is noise, we change z to the average of the z_i ; in other words, in choosing the new gray level for the point, we pay no attention to its old level. On the other hand, if we decide that the point is not noise, we do not change z at all, i.e., we pay no further attention to its neighbors’ levels. A more general approach would be to make a “fuzzy” decision about the point, say that it has probability p of being a noise point. We could then give the point a new gray level which would be a weighted average of z and the z_i ’s, e.g., $(1-p)z + p \sum z_i/k$. If p is high, so that the point is highly likely to be a noise point, this formula gives high weight to the neighborhood average and low weight to the point’s old gray level; while if p is low, the opposite is true.

If the average over a large neighborhood is used to replace the gray level of a suspected noise point, more weight should normally be given to nearby neighbors than to distant neighbors. Alternatively, one could determine the proper gray level to interpolate at the point by fitting some type of surface to the gray levels of the neighboring points.

It should be remembered that averaging, or any other operation based on the neighborhood of a point, requires careful definition when the point is on or near the border of the picture. This problem is especially troublesome when large neighborhoods are used. One can, of course, give the operation a complex definition to cover the cases of points near the picture border. As a simpler

alternative, one can define the operation uniformly throughout the picture, treating neighborhood points that lie outside the picture as having value zero (say), and then ignoring the results near the border as not meaningful.

6.5.2 Averaging

The smoothing methods described in the preceding section depend on first distinguishing noise from nonnoise in the picture, then removing the noise and “mending” the picture by interpolation. In this section, we discuss another class of smoothing methods which do not depend on identification and removal of the noise. Instead, these methods weaken the noise by applying some type of averaging to the picture. Since averaging blurs the picture, these methods too must be applied with care, to avoid degrading sharp detail such as lines or edges.

To see why averaging can weaken noise, suppose that the noise values at each point are independent samples chosen from a distribution having mean 0 and standard deviation σ . Suppose also that we are averaging over a set of picture points whose nonnoisy gray levels are $z_1, \dots, z_n$, and let the noise values at these points be $w_1, \dots, w_n$. Then in the average, $(z_1 + \dots + z_n)/n + (w_1 + \dots + w_n)/n$, the second term can be regarded (see Section 2.3.5, Exercise 6) as a sample of a random variable with mean 0 and standard deviation $\sigma/\sqrt{n}$. Thus by averaging, we have reduced the amplitude of the noise fluctuations.

We can perform the averaging without any danger of blurring when we are given several independently noisy copies $f_i(x, y)$ of a picture, e.g., several grainy photographs, or several “snowy” TV frames, of the same scene, where the noise values of the copies at a given point are all independent. In this case, we can reduce the noise by pointwise averaging the copies, obtaining the new picture f defined by

$$f(x, y) = (1/n) \sum_{i=1}^n f_i(x, y).$$

An example of smoothing by averaging of independent copies is shown in Fig. 18.

The method of pointwise averaging can also be used when the picture to be smoothed is symmetric or periodic (except for the noise). Specifically, let P be a pattern or region that occurs several times on the picture; let $P_1, \dots, P_n$ be the noisy instances of P . If we average these instances pointwise, we obtain a less noisy version $\bar{P}$ of P , and we can replace each of the P_i ’s by $\bar{P}$ to obtain a smoother picture.

If we only have a single noisy picture available, we can attempt to reduce the

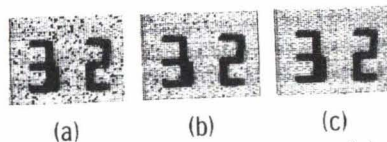


Fig. 18 Smoothing by averaging over different instances of the noise. (a)–(c) Averages of 2, 4, and 8 independently noisy versions of Fig. 17a.

noise level by *local* averaging, i.e., we give each point a new gray level which is the average of the original gray levels in some neighborhood of the point (including the point itself). This is relatively straightforward if the noise is finer grained than the smallest picture detail of interest; it should then be possible, by averaging over a small neighborhood of every point, to reduce the noise while keeping the blur at a negligible, or at least tolerable, level. Alternatively, we may be able to remove the noise by suppressing or deemphasizing high spatial frequencies—or equivalently, emphasizing low spatial frequencies—in the picture’s Fourier transform. This last approach has been successfully used to remove film-grain noise from photographs.

If the given picture is two-valued, noise that is smaller than the picture detail can be removed by a process of shrinking and reexpanding. For example, suppose that we change all black points to white if they have any white neighbors, and then change all white points to black if they have any black neighbors. The first step shrinks all black regions, while the second step reexpands them; but a black object that is two points wide or less will disappear completely at the first step, so that the second step cannot restore it. Thus this process deletes not only isolated points, but also thin lines. Other uses of this shrinking and expanding operations will be discussed in Sections 9.2.4 and 9.2.5.

Local averaging and low-emphasis filtering cannot be used indiscriminately if the picture contains sharp edges or fine lines, since these involve high spatial frequencies. One can, however, use some sort of *selective* local averaging in which the presence of edges and lines is detected and the averaging is done only at points where they are not present. (Edge and line detection techniques are discussed in Sections 8.2 and 8.3.) As a refinement of this idea, when an edge (or line) is present at a point, one can take a directional average at that point, involving only those neighbors of the point that lie in the direction along the edge; while if no edge is present, one can average isotropically, using neighbors on all sides of the point.

The size of the neighborhood that should be used in local averaging depends on the degree of noise reduction that is desired; as pointed out earlier, averaging n points should reduce the standard deviation of the noise by a factor of $\sqrt{n}$. If the noise is signal dependent, one may want to use a different

neighborhood size at each point, depending on the average picture gray level in the vicinity of that point, in order to obtain a uniform reduced noise level throughout the picture. One way of obtaining such variable-sized neighborhoods is to calculate the sums of the gray levels in a series of expanding neighborhoods of a point, and stop expanding when the sum reaches a preset threshold. The resulting neighborhood will be large if the gray levels in the vicinity are generally low, and vice versa. The average gray level over the neighborhood is then inversely proportional to the neighborhood’s area, since the sum of gray levels over the neighborhood is a constant. A variable-neighborhood averaging method of this type has been used to smooth “quantum-limited” pictures, which consist of clusters of dots; here the neighborhood size at a point was determined by expanding the neighborhood until it contained a preset number of dots.

Whether the neighborhood size used is fixed or variable, it would be desirable for the neighborhood not to cross over edges in the picture, and not to contain lines, since we want the neighborhood average to represent a “uniform” piece of the picture around the given point. We can “grow” such a neighborhood around the point by accepting neighboring points if their gray levels do not differ too greatly from that of the point or of each other; then accepting neighbors of these neighbors if their levels are not too different from the average level of the already accepted points; and so on (see Section 8.4.3 on such methods of “region growing”). Here again, “too different” can be defined in terms of the standard deviation of the noise. The growing neighborhood will hopefully exclude high-contrast edges, lines, and points, since these will differ too greatly from the surrounding average; it may thus have a highly irregular shape. The growth process can be stopped after a specified number of steps, if desired. This method is applicable only in homogeneous regions of the nonnoisy picture; it would be much more complicated to generalize it to textured regions.

Averaging that avoids edges (or lines) will be less effective in reducing the noise level near edges. Noise is, however, generally less conspicuous when it is located near edges; hence this limitation may not be serious.

A selective averaging technique can also be used to remove quantization noise from a picture. If a picture has been quantized too coarsely, i.e., to a set of levels that are spaced too far apart (say Δ apart) and are thus easily discriminable, the picture will contain conspicuous “false contours” separating regions that have different levels (see Section 4.3). We can remove the false contours by averaging the picture and quantizing the averages to a larger set of levels, closer together than Δ . To avoid blurring real edges in the picture, we can suppress the averaging at edges where the gray level changes by a multiple of Δ ; but we allow averaging at edges where the change is only Δ , since such edges are likely to be false contours.

Another way to remove false contours is to add noise (of magnitude less than Δ) to the picture, again using a finer set of quantization levels; this will tend to break up the false contours. This last approach is essentially the same as the Roberts compression technique mentioned in Section 5.3. In effect, it trades quantization noise for salt-and-pepper noise.

6.6 BIBLIOGRAPHICAL NOTES

For general reviews of image quality and enhancement see Linfoot [18], Huang [12], Levi [17], McCamy [20], Prewitt [27], and Biberman [4]. Two recent discussions of the use of visual system models in the design of image-processing systems, particularly as regards the nonlinear brightness response of the eye and its dependence on spatial and temporal frequency, are Budrikis [5] and Stockham [29]. The space-domain effects of frequency-domain errors are treated by Anderson and Huang [2]. On histogram modification see Hall *et al.* [10], Troy *et al.* [31], and Haralick *et al.* [11]; see also the references on tapered and optimum quantization in Section 4.4.

Much work on image enhancement, including both gray scale and geometric correction techniques, as well as filtering techniques for sharpening and smoothing, has been done at NASA's Jet Propulsion Laboratory (see, e.g., O'Handley and Green [23]). For further discussion of geometrical operations on digital pictures see Johnston and Rosenfeld [13].

The use of derivative operations for sharpening a picture is very old (e.g., Goldmark and Hollywood [7]; Kovasznay and Joseph [15, 16]). A more recent review by Prewitt [27] also discusses Laplacians and their modifications (those based on the second derivative in the gradient direction are due to Gabor). Transfer function modification by high-emphasis filtering is also a classical technique; for a recent discussion see Arguello *et al.* [3]. Regarding filtering based on a logarithmic transformation, see Oppenheim *et al.* [24] and Stockham [29]. On the use of derivative operations for edge and line detection see Sections 8.2 and 8.3.

A global approach to smoothing, using mathematical programming, is discussed by Martelli and Montanari [19]. Techniques for removing "salt-and-pepper" noise (in the two-valued case) are described by Dinneen [6]. On smoothing by averaging multiple copies of an image see Kohler and Howell [14]. The application to symmetric objects is given by Agrawal *et al.* [1] and Norman [22]. For early work on smoothing of film-grain noise see Thiry [30]. On noise removal by shrinking and reexpanding see reference [35] in Section 9.5. Regarding averaging in the absence of, or along, edges and lines, see Graham [9]; for a generalization to texture edges see Newman and Dirilten [21]. Variable-neighborhood averaging is discussed by Pizer and Vetter [25, 26] for the case of quantum-limited pictures; they also describe

methods of deblurring such pictures by shifting dots in the direction for which local dot density is increasing fastest. Averaging based on region growing is treated by Roetling [28].

A major topic not treated here is that of reconstructing pictures from their one-dimensional projections, and three-dimensional objects from their two-dimensional projections. A good review of methods for solving the latter problem, for the case of nonopaque objects, is given by Gordon and Herman [8]. The related topic of estimating point spread functions from line spread functions is treated in Section 7.1.2, where additional references are given.

REFERENCES

1. H. O. Agrawal, J. W. Kent, and D. M. MacKay, Rotation technique in electron microscopy of viruses, *Science* **148**, 1965, 638-640.
2. G. B. Anderson and T. S. Huang, Frequency-domain image errors, *Pattern Recognit.* **3**, 1961, 185-196.
3. R. J. Arguello, H. R. Sellner, and J. Z. Stuller, Transfer function compensation of sampled imagery, *IEEE Trans. Comput.* **C-21**, 1972, 812-818.
4. L. M. Biberman (ed.), "Perception of Displayed Information." Plenum Press, New York, 1973.
5. Z. L. Budrikis, Visual fidelity criterion and modeling, *Proc. IEEE* **60**, 1972, 771-779.
6. G. P. Dinneen, Programming pattern recognition, *Proc. Western Joint Comput. Conf.* 1955, 94-100.
7. P. C. Goldmark and J. M. Hollywood, A new technique for improving the sharpness of television pictures, *Proc. IRE* **39**, 1951, 1314-1322.
8. R. Gordon and G. T. Herman, Three-dimensional reconstruction from projections: a review of algorithms, *Int. Rev. Cytol.* **38**, 1974, 111-151.
9. R. E. Graham, Snow removal—a noise-stripping process for picture signals, *IRE Trans. Informat. Theory* **IT-8**, 1962, 129-144.
10. E. L. Hall, R. P. Kruger, S. J. Dwyer III, D. L. Hall, R. W. McLaren, and G. S. Lodwick, A survey of preprocessing and feature extraction techniques for radiographic images, *IEEE Trans. Comput.* **C-20**, 1971, 1032-1044.
11. R. M. Haralick, K. Shanmugam, and I. Dinstein, Textural features for image classification, *IEEE Trans. Syst. Man Cybernet.* **SMC-3**, 1973, 610-621.
12. T. S. Huang, Image enhancement: a review, *Opto-Electron.* **1**, 1969, 49-59.
13. E. G. Johnston and A. Rosenfeld, Geometrical operations on digitized pictures, in "Picture Processing and Psychopictorics" (B. S. Lipkin and A. Rosenfeld, eds.), pp. 217-240. Academic Press, New York, 1970.
14. R. Kohler and H. Howell, Photographic image enhancement by superimposition of multiple images, *Photogr. Sci. Eng.* **7**, 1963, 241-245.
15. L. S. G. Kovasznay and H. M. Joseph, Processing of two-dimensional patterns by scanning techniques, *Science* **118**, 1953, 475-477.
16. L. S. G. Kovasznay and H. M. Joseph, Image processing, *Proc. IRE* **43**, 1955, 560-570.
17. L. Levi, On image evaluation and enhancement, *Opt. Acta* **17**, 1970, 59-76.
18. E. H. Linfoot, "Fourier Methods in Optical Image Evaluation." Focal Press, New York, 1964.
19. A. Martelli and U. Montanari, Optimal smoothing in picture processing: an application to fingerprints, *Proc. IFIP Congr.* **71** Booklet TA-2, 86-90.

20. C. S. McCamy, The evaluation and manipulation of photographic images, in "Picture Processing and Psychopictorics" (B. S. Lipkin and A. Rosenfeld, eds.), pp. 57-74. Academic Press, New York, 1970.
21. T. G. Newman and H. Dirilten, A nonlinear transformation for digital picture processing, *IEEE Trans. Comput.* **C-22**, 1973, 869-873.
22. R. S. Normar, Rotation technique in radially symmetric electron micrographs: mathematical analysis, *Science* **152**, 1966, 1238-1239.
23. D. A. O'Handley and W. B. Green, Recent developments in digital image processing at the Image Processing Laboratory at the Jet Propulsion Laboratory, *Proc. IEEE* **60**, 1972, 821-828.
24. A. V. Oppenheim, R. W. Schafer, and T. G. Stockham, Jr., Nonlinear filtering of multiplied and convolved signals, *Proc. IEEE* **56**, 1968, 1264-1291.
25. S. M. Pizer and H. G. Vetter, Perception and processing of medical radio-isotope scans, in "Pictorial Pattern Recognition" (G. C. Cheng et al., eds.), pp. 147-156. Thompson, Washington, D. C., 1968.
26. S. M. Pizer and H. G. Vetter, Processing quantum-limited images, in "Picture Processing and Psychopictorics" (B. S. Lipkin and A. Rosenfeld, eds.), pp. 165-176. Academic Press, New York, 1970.
27. J. M. S. Prewitt, Object enhancement and extraction, in "Picture Processing and Psychopictorics" (B. S. Lipkin and A. Rosenfeld, eds.), pp. 75-149. Academic Press, New York, 1970.
28. P. G. Roetling, Image enhancement by noise suppression, *J. Opt. Soc. Amer.* **60**, 1970, 867-869.
29. T. G. Stockham, Jr., Image processing in the context of a visual model, *Proc. IEEE* **60**, 1972, 828-842.
30. H. Thiry, Some qualitative and quantitative results on spatial filtering of granularity, *Appl. Opt.* **3**, 1964, 39-43.
31. E. B. Troy, E. S. Deutsch, and A. Rosenfeld, Gray-level manipulation experiments for texture analysis, *IEEE Trans. Syst. Man Cybernet.* **SMC-3**, 1973, 91-98.

Chapter 7

Restoration

Picture restoration deals with images that have been recorded in the presence of one or more sources of degradation. There are many sources of degradation in imaging systems. Some types of degradation affect only the gray levels of the individual picture points, without introducing spatial blur; these are sometimes called *point degradations*. Other types which do involve blur are called *spatial degradations*. Still other types involve chromatic or temporal effects. In this chapter, we will only be concerned with point and spatial degradations. Such degradations occur in a variety of applications. In aerial reconnaissance, astronomy, and remote sensing the pictures one obtains are degraded by atmospheric turbulence, aberrations of the optical system, and relative motion between the camera and the object. Electron micrographs are often degraded by the spherical aberration of the electron lens. Medical radiographic images are of low resolution and contrast due to the nature of x-ray imaging systems.

Given an ideal picture $f(x, y)$ and the corresponding degraded picture $g(x, y)$, we will assume that g and f are related by

$$g(x, y) = \iint h(x, y, x', y') f(x', y') dx' dy' + v(x, y) \quad (1)$$

where $h(x, y, x', y')$ is the degradation function and $v(x, y)$ the random noise that may be present in the output picture. Evidently, in the absence of noise the degraded image of a point source described by $f(x', y') = \delta(x' - \alpha, y' - \beta)$