Computer Science
and Applied Mathematics

DIGITAL PICTURE PROCESSING

SECOND EDITION

Azriel Rosenfeld and Avinash C. Kak

Volume 1

Digital Picture Processing

Second Edition

Volume 1

AZRIEL ROSENFELD

Computer Science Center
University of Maryland
College Park, Maryland

AVINASH C. KAK

School of Electrical Engineering
Purdue University
West Lafayette, Indiana

1982

ACADEMIC PRESS
A Subsidiary of Harcourt Brace Jovanovich, Publishers

New York London
Paris San Diego San Francisco São Paulo Sydney Tokyo Toronto

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 7DX

Library of Congress Cataloging in Publication Data

Rosenfeld, Azriel.
Digital picture processing.

(Computer science and applied mathematics)
Includes bibliographical references and index.
1. Optical data processing. I. Kak, Avinash C.
II. Title. III. Series.
TA1630.R67 1981 621.3819'598 81-17611
ISBN 0-12-597301-2 AACR2

PRINTED IN THE UNITED STATES OF AMERICA

82 83 84 85 9 8 7 6 5 4 3 2 1

Preface

The rapid rate at which the field of digital picture processing has grown in the past five years has necessitated extensive revisions and the introduction of topics not found in the original edition.

Two new chapters have been added: Chapter 8 (by A. C. K.) on reconstruction from projections and Chapter 9 (by A. R.) on matching. The latter includes material from Chapters 6 and 8 of the first edition on geometric transformations and matching, but it consists primarily of new material on imaging geometry, rectification, and stereomapping, as well as an appendix on the analysis of time-varying imagery.

Chapter 2 incorporates a new section on vector space representation of images. Chapter 5 on compression has undergone a major expansion. It includes a new section on interpolative representation of images and fast implementation of the Karhunen–Loève transform based thereon. Also included in this chapter is image compression using discrete cosine transforms—a technique that has attracted considerable attention in recent years. New sections on block quantization, the recently discovered technique of block truncation compression, and error-free compression have also been added to Chapter 5. New material has been added to Chapter 6 on gray level and histogram transformation and on smoothing. Chapter 7 has also been considerably expanded and includes many new restoration techniques. This chapter incorporates a new frequency domain derivation of the constrained least squares filter. The treatment of Markov representation has been expanded with a section on vector–matrix formulation of such representations. Chapters 10, 11, and 12 are major expansions of the first edition's Chapters 8–10, dealing with segmentation of pictures into parts, representations of the parts (formerly "geometry"), and description of pictures in terms of parts. Chapter 10 incorporates much new material on pixel classification, edge detection, Hough transforms, and picture partitioning, reflecting recent developments in these areas; it also contains an entirely new section (10.5) on iterative "relaxation" methods for fuzzy or probabilistic segmentation. Chapter 11 is now organized according to types of representations (runs, maximal blocks, quadtrees, border codes), and discusses how to convert between these representations and how to use them to compute geometrical properties of picture subsets. Chapter 12 treats picture properties as well as descriptions of pictures at various levels (numerical arrays, region representations, relational structures). It also discusses models for classes of pictures, as defined, in particular, by constraints that must be satisfied at a given level of description ("declarative models") or by grammars that generate or accept the classes. It considers how to construct a model consistent with a given set of descriptions and how to extract a description that matches a given model; it also contains an appendix on the extraction of three-dimensional information about a scene from pictures.

The size of this second edition has made it necessary to publish this book in two volumes. However, a single chapter numbering has been maintained. Volume 1 contains Chapters 1–8, covering digitization, compression, enhancement, restoration, and reconstruction; and Volume 2 contains Chapters 9–12, covering matching, segmentation, representation, and description. The material in Volume 2 is not strongly dependent on that in Volume 1; and to make it even more self-contained, the Preface and Introduction (called Chapter 1 in Volume 1) are reproduced at the beginning of Volume 2.

The authors of the chapters are as follows: Chapters 2, 4, 5, 7, and 8 are by A. C. K, whereas Chapters 1, 3, 6, and 9 through 12 are by A. R.

Acknowledgments

A. C. K. would like to express appreciation to his colleague O. Robert Mitchell and his graduate students David Nahamoo, Carl Crawford, and Kai-Wah Chan for many stimulating discussions. A number of our colleagues helped us by providing figures (see the individual figure credits). In addition, many of the figures were produced by Andrew Pilipchuk and others at the University of Maryland; and by Carl Crawford, Doug Morton, and Kai-Wah Chan at Purdue University. Kathryn Riley, Dawn Shifflett, and Mellanie Boes, among others, did an excellent job of preparing the text. To these, and to others too numerous to mention, our sincerest thanks.

The authors wish to express their indebtedness to the following individuals and organizations for their permission to reproduce the figures listed below.

Chapter 3: Figures 1, 4–7, and 11, from T. N. Cornsweet, *Visual Perception,* Academic Press, New York, 1970. Figure 3, from J. Beck, *Surface Color Perception,* copyright © 1972 by Cornell University; used by permission of Cornell University Press, Ithaca, New York. Figure 8, from S. Coren, "Subjective contours and apparent depth," *Psychological Review* **79,** 1972, 359–367. Figures 9–10, from M. Luckiesh, *Visual Illusions,* Dover, New York, 1965. Figure 12, from J. Beck, "Similarity grouping and peripheral discriminability under uncertainty," *American*

Journal of Psychology **85,** 1972, 1–19. Figures 13–16, from M. Wertheimer, "Principles of perceptual organization," in D. C. Beardslee and M. Wertheimer, eds., *Readings in Perception,* pp. 115–135, © 1958 by Litton Educational Publishing, Inc.; reprinted by permission of Van Nostrand Reinhold Company, Princeton, New Jersey. Figures 17–19, from J. Hochberg, *Perception,* © 1964, by permission of Prentice-Hall, Inc., Englewood Cliffs, New Jersey.

Chapter 4: Figure 8, from R. Legault, "The aliasing problems in two-dimensional sampled imagery," in L. M. Biberman, ed., *Perception of Displayed Information,* 279–312, Plenum Press, New York, 1973. Figure 10, from L. M. Biberman, "A summary," *ibid.,* 313–322. Figure 12, from A. Habibi and P. A. Wintz, "Image coding by linear transformation and block quantization," *IEEE Transactions on Communication Technology* **COM-19,** 1971, 50–62. Figure 15, from T. S. Huang, O. J. Tretiak, B. Prasada, and Y. Yamaguchi, "Design considerations in PCM transmission of low-resolution monochrome still pictures," *Proceedings of the IEEE* **55,** 1967, 331–335. Figures 17–19, from T. S. Huang, "PCM picture transmission," *IEEE Spectrum* **2,** 1965, 57–63. Table 1, from J. Max, "Quantizing for minimum distortion," *IRE Transactions on Information Theory* **IT–6,** 1960, 7–12.

Chapter 5: Figures 1 and 3–9, from P. A. Wintz, "Transform picture coding," *Proceedings of the IEEE* **60,** 1972, 809–820. Figures 16–18 and Tables 2–3, from A. Habibi, "Comparison of *n*th-order DPCM encoder with linear transformations and block quantization techniques," *IEEE Transactions on Communication Technology* **COM-19,** 1971, 948–956. Figures 21 and 22 from E. J. Delp and O. R. Mitchell, "Image compression using block truncation coding," *IEEE Transactions on Communication Technology* **COM-27,** 1979, 1335–1342.

Chapter 6: Figure 14, 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," *Proceedings of the IEEE* **60,** 1972, 821–828. Figure 15, from T. G. Stockham, Jr., "Image processing in the context of a visual model," *ibid.,* 828–842.

Chapter 7: Figure 3, from J. L. Harris, Sr., "Image evaluation and restoration," *Journal of the Optical Society of America* **56,** 1966, 569–574. Figure 4, from B. L. McGlamery, "Restoration of turbulence-degraded images," *ibid.* **57,** 1967, 293–297. Figure 5, from J. L. Harris, Sr., "Potential and limitations of techniques for processing linear motion-degraded imagery," in *Evaluation of Motion-Degraded Images,* pp. 131–138, NASA SP-193, December 1968. Figure 6, from B. R. Hunt, "The application of constrained least-squares estimation to image restoration by digital computer," *IEEE Transactions on Computers* **C-22,** 1973, 805–812.

Contents

Chapter 1

Introduction

1.1 PICTURE PROCESSING

Picture processing or *image processing* is concerned with the manipulation and analysis of pictures by computer. Its major subareas include

(a) *Digitization and compression*: Converting pictures to discrete (digital) form; efficient coding or approximation of pictures so as to save storage space or channel capacity.
(b) *Enhancement, restoration, and reconstruction*: Improving degraded (low-contrast, blurred, noisy) pictures; reconstructing pictures from sets of projections.
(c) *Matching, description, and recognition*: Comparing and registering pictures to one another; segmenting pictures into parts, measuring properties of and relationships among the parts, and comparing the resulting descriptions to models that define classes of pictures.

In this chapter we introduce some basic concepts about pictures and digital pictures, and also give a bibliography of general references on picture processing and recognition. (References on specific topics are given at the end of each chapter.) Chapter 2 reviews some of the mathematical tools used in later chapters, including linear systems, transforms, and random fields, while Chapter 3 briefly discusses the psychology of visual perception.

The remaining chapters deal with the theory of digitization (4); coding and compression (5); enhancement (6); restoration and estimation (7); reconstruction from projections (8); registration and matching (9); segmentation into parts (10); representation of parts and geometric property measurement (11); and nongeometric properties, picture descriptions, and models for classes of pictures (12).

The level of treatment emphasizes concepts, algorithms, and (when necessary) the underlying theory. We do not cover hardware devices for picture input (scanners), processing, or output (displays); nondigital (e.g., optical) processing; or picture processing software.

1.2 SCENES, IMAGES, AND DIGITAL PICTURES

1.2.1 Scenes and Images

When a scene is viewed from a given point, the light received by the observer varies in brightness and color as a function of direction. Thus the information received from the scene can be expressed as a function of two variables, i.e., of two angular coordinates that determine a direction. (The scene brightness and color themselves are resultants of the illumination, reflectivity, and geometry of the scene; see Section 6.2.2.)

In an optical image of the scene, say produced by a lens, light rays from each scene point in the field of view are collected by the lens and brought together at the corresponding point of the image. Scene points at different distances from the lens give rise to image points at different distances; the basic equation is

$$\frac{1}{u} + \frac{1}{v} = \frac{1}{f}$$

where u, v are the distances of the object and image points from the lens (on opposite sides), and f is a constant called the focal length of the lens. If u is large, i.e., the scene points are all relatively far from the lens, $1/u$ is negligible, and we have $v \approx f$, so that the image points all lie at approximately the same distance from the lens, near its "focal plane." Thus the imaging process converts the scene information into an illumination pattern in the image plane; this is still a function of two variables, but they are now coordinates in the plane. (Image formation by optical systems will not be further discussed here. On the geometry of the mapping from three-dimensional scene coordinates to two-dimensional image coordinates, see Section 9.1.2.)

We can now record or measure the pattern of light from the scene by placing some type of sensor in the image plane. (Some commonly used sensors will be mentioned in the next paragraph.) Any given sensor has a characteristic spectral sensitivity, i.e., its response varies with the color of the light; thus its total response to the light at a given point can be expressed by an integral of the form $\int S(\lambda)I(\lambda)\,d\lambda$, where $I(\lambda)$ is light intensity and $S(\lambda)$ is sensitivity as functions of wavelength. This means that if we use only a single sensor, we can only measure (weighted) light intensity. If we want to measure color, we must use several sensors having different spectral responses; or we must split the light into a set of spectral bands, using color filters, and measure the light intensity in each band. (Knowing the intensities in three suitably chosen bands, e.g., in the red, green, and blue regions of the spectrum, is enough to characterize any color; see Section 3.3.) In other words, when we use only one sensor, we are representing the scene information by a scalar-valued function of position in the image, representing scene brightness. To represent color, we use a k-tuple (usually a triple) of such functions, or equivalently, a vector-valued function, representing the brightness in a set of spectral bands. We will usually assume in this book that we are dealing with a single scalar-valued brightness function. Photometric concepts and terminology will not be further discussed here; we use terms such as "brightness" and "intensity" in an informal sense.

Image sensors will not be discussed in detail in this book, but we briefly mention here some of the most common types.

(a) We can put an *array* of photosensitive devices in the image plane; each of them measures the scene brightness at a particular point (or rather, the total scene brightness in a small patch).

(b) We need only a single photosensor in the image plane if we can illuminate the scene one point (or small patch) at a time; this is the principle of the *flying-spot scanner*. Similarly, we need only one photosensor if we can view the scene through a moving aperture so that, at any given time, the light from only one point of the scene can reach the sensor.[§]

(c) In a *TV camera*, the pattern of brightness in the scene is converted into an electrical charge pattern on a grid; this pattern can then be scanned by an electron beam, yielding a *video signal* whose value at any given time corresponds to the brightness at a given image point.

In all of these schemes, the image brightness is converted into a pattern of electrical signals, or into a time-varying signal corresponding to a sequential

[§] As a compromise between (a) and (b), we can use a one-dimensional array of sensors in the image plane, say in the horizontal direction, and scan in the vertical direction, so that light from only one "row" of the scene reaches the sensors at any given time.

scan of the image or scene. Thus the sensor provides an electrical or electronic analog of the scene brightness function, which is proportional to it, if the sensors are linear. More precisely, an array sensor provides a discrete array of samples of this function; while scanning sensors provide a set of cross sections of the function along the lines of the scanning pattern.

If, instead of using a sensor, we put a piece of photographic film (or some other light-sensitive recording medium) in the image plane, the brightness pattern gives rise to a pattern of variations in the optical properties of the film. (Color film is composed of layers having different spectral sensitivities; we will discuss here only the black-and-white case.) In a film transparency, the optical transmittance t (i.e., the fraction of the light transmitted by the film) varies from point to point; in an opaque print, the reflectance r (= the fraction of light reflected) varies. Evidently we have $0 \leqslant t \leqslant 1$ and $0 \leqslant r \leqslant 1$. The quantity $-\log t$ or $-\log r$ is called optical *density*; thus a density close to zero corresponds to almost perfect transmission or reflection, while a very high density, say 3 or 4, corresponds to almost perfect opaqueness or dullness (i.e., only 10^{-3} or 10^{-4} of the incident light is transmitted or reflected). For ordinary photographic processes, the density is roughly a linear function of the log of the amount of incident light (the log of the "exposure") over a range of exposures; the slope of this line is called photographic *gamma*. Photographic processes will not be discussed further in this book. A photograph of a scene can be converted into signal form by optically imaging it onto a sensor.

1.2.2 Pictures and Digital Pictures

We saw in the preceding paragraphs that the light received from a scene by an optical system produces a two-dimensional image. This image can be directly converted into electrical signal form by a sensor, or it can be recorded photographically as a picture and subsequently converted. Mathematically, a *picture* is defined by a function $f(x, y)$ of two variables (coordinates in the image plane, corresponding to spatial directions). The function values are brightnesses, or k-tuples of brightness values in several spectral bands. In the black-and-white case, the values will be called *gray levels*. These values are real, nonnegative (brightness cannot be negative), and bounded (brightness cannot be arbitrarily great). They are zero outside a finite region, since an optical system has a bounded field of view, so that the image is of finite size; without loss of generality, we can assume that this region is rectangular. Whenever necessary, we will assume that picture functions are analytically well-behaved, e.g., that they are integrable, have invertible Fourier transforms, etc.

When a picture is digitized (see Chapter 4), a *sampling* process is used to extract from the picture a discrete set of real numbers. These *samples* are usually the gray levels at a regularly spaced array of points, or, more realistically, average gray levels taken over small neighborhoods of such points. (On other methods of sampling see Section 4.1.) The array is almost always taken to be Cartesian or rectangular, i.e., it is a set of points of the form (md, nd), where m and n are integers and d is some unit distance. (Other types of regular arrays, e.g., hexagonal or triangular, could also be used; see Section 11.1.7, Exercise 4, on a method of defining a hexagonal array by regarding alternate rows of a rectangular array as shifted $d/2$ to the right.) Thus the samples can be regarded as having integer coordinates, e.g., $0 \leqslant m < M, 0 \leqslant n < N$.

The picture samples are usually *quantized* to a set of discrete gray level values, which are often taken to be equally spaced (but see Section 4.3). In other words, the *gray scale* is divided into equal intervals, say $I_0, \ldots, I_K$, and the gray level $f(x, y)$ of each sample is changed into the level of the midpoint of the interval I_i in which $f(x, y)$ falls. The resulting quantized gray levels can be represented by their interval numbers $0, \ldots, K$, i.e., they can be regarded as integers.

The result of sampling and quantizing is a *digital picture*. As just seen, we can assume that a digital picture is a rectangular array of integer values. An element of a digital picture is called a *picture element* (often abbreviated *pixel* or *pel*); we shall usually just call it a *point*. The value of a pixel will still be called its gray level. If there are just two values, e.g., "black" and "white," we will usually represent them by 0 and 1; such pictures are called two-valued or binary-valued.

Digital pictures are often very large. For example, suppose we want to sample and quantize an ordinary (500-line) television picture finely enough so that it can be redisplayed without noticeable degradation. Then we must use an array of about 500 by 500 samples, and we should quantize each sample to about 50 discrete gray levels, i.e., to about a 6-bit number. This gives us an array of 250,000 6-bit numbers, for a total of $1\frac{1}{2}$ million bits. In many cases, even finer sampling is necessary; and it has become standard to use 8-bit quantization, i.e., 256 gray levels.

Except on the borders of the array, any point (x, y) of a digital picture has four horizontal and vertical neighbors and four diagonal neighbors, i.e.,

$$\begin{array}{lll} (x-1, y+1) & (x, y+1) & (x+1, y+1) \\ (x-1, y) & (x, y) & (x+1, y) \\ (x-1, y-1) & (x, y-1) & (x+1, y-1) \end{array}$$

In this illustration of the 3×3 *neighborhood* of a point we have used Cartesian coordinates (x, y), with x increasing to the right and y increasing upward. There are other possibilities; for example, one could use matrix coordinates (m, n), in which m increases downward and n to the right. Note that the diagonal neighbors are $\sqrt{2}$ units away from (x, y), while the horizontal and vertical neighbors are only one unit away. If we think of a pixel as a unit square, the horizontal and vertical neighbors of (x, y) share a side with (x, y), while its diagonal neighbors only touch it at a corner. Some of the complications introduced by the existence of these two types of neighbors will be discussed in Chapter 11. Neighborhoods larger than 3×3 are sometimes used; in this case, a point may have many types of neighbors.

If (x, y) is on the picture border, i.e., $x = 0$ or $M - 1$, $y = 0$ or $N - 1$, some of its neighbors do not exist, or rather are not in the picture. When we perform operations on the picture, the new value of (x, y) often depends on the old values of (x, y) and its neighbors. To handle cases where (x, y) is on the border, we have several possible approaches:

(a) We might give the operation a complex definition that covers these special cases. However, this may not be easy, and in any case it is computationally costly.

(b) We can regard the picture as cyclically closed, i.e., assume that column $M - 1$ is adjacent to column 0 and row $N - 1$ to row 0; in other words, we take the coordinates (x, y) modulo (M, N). This is equivalent to regarding the picture as an infinite periodic array with an $M \times N$ period. We will sometimes use this approach, but it is usually not natural, since the opposite rows and columns represent parts of the scene that are not close together.

(c) We can assume that all values outside the picture are zero. This is a realistic way of representing the image (see the first paragraph of this section), but not the scene.

(d) The simplest approach is to apply the operation only to a *subpicture*, chosen so that for all (x, y) in the subpicture, the required neighbors exist in the picture. This yields results all of which are meaningful; but note that the output picture produced by the operation is smaller than the input picture.

1.2.3 Operations on Pictures

In this book we shall study many different types of operations that can be performed on digital pictures to produce new pictures. The following are some of the important types of picture operations:

(a) *Point operations*: The output gray level at a point depends only on the input gray level at the same point. Such operations are extensively used for gray scale manipulation (Section 6.2) and for segmentation by pixel classification (Section 10.1). There may be more than one input picture; for example, we may want to take the difference or product of two pictures, point by point. In this case, the output level at a point depends only on the set of input levels at the same point.

(b) *Local operations*: The output level at a point depends only on the input levels in a neighborhood of that point. Such operations are used for deblurring (Section 6.3), noise cleaning (Section 6.4), and edge and local feature detection (Sections 10.2 and 10.3), among other applications.

(c) *Geometric operations*: The output level at a point depends only on the input levels at some other point, defined by a geometrical transformation (e.g., translation, rotation, scale change, etc.) or in a neighborhood of that point. On such operations see Section 9.3.

An operation $\mathcal{O}$ is called *linear* if we get the same output whether we apply $\mathcal{O}$ to a linear combination of pictures (i.e., we take $\mathcal{O}(af + bg)$) or we apply $\mathcal{O}$ to each of the pictures and then form the same linear combination of the results (i.e., $a\mathcal{O}(f) + b\mathcal{O}(g)$). Linear operations on pictures will be discussed further in Section 2.1.1. Point and local operations may or may not be linear. For example, simple stretching of the gray scale ($\mathcal{O}(f) = cf$) is linear, but thresholding ($\mathcal{O}(f) = 1$ if $f \geqslant t$, $=0$ otherwise) is not; local averaging is linear, but local absolute differencing is not. Geometric operations are linear, if we ignore the need to redigitize the picture after they are performed (Section 9.3).

$\mathcal{O}$ is called *shift invariant* if we get the same output whether we apply $\mathcal{O}$ to a picture and then shift the result, or first shift the picture and then apply $\mathcal{O}$. Such operations will be discussed further in Section 2.1.2. The examples of point and local operations given in the preceding paragraph are all shift invariant, but we can also define shift variant operations of these types, e.g., modifying the gray level of a point differently, or taking a different weighted average, as a function of position in the picture. The only shift-invariant geometric operations are the shifts, i.e., the translations. It is shown in Section 2.1.2 that an operation is linear and shift invariant iff it is a *convolution*; this is an operation in which the output gray level at a point is a linear combination of the input gray levels, with coefficients that depend only on their positions relative to the given point, but not on their absolute positions.

In Chapters 11 and 12 we will discuss *picture properties*, i.e., operations that can be performed on pictures to produce numerical values. In particular, we will deal with *point* and *local properties* (whose values depend only on one point, or on a small part, of the picture); *geometric properties* of picture subsets (whose values depend only on the set of points belonging to the given subset, but not on their gray levels); and *linear properties* (which give