Mathematics of Digital Images

Creation, Compression, Restoration, Recognition

S. G. HOGGAR

CAMBRIDGE

CAMBRIDGE www.cambridge.org/9780521780292

This page intentionally left blank

MATHEMATICS OF DIGITAL IMAGES

Creation, Compression, Restoration, Recognition

Compression, restoration and recognition are three of the key components of digital imaging. The mathematics needed to understand and carry out all these components is here explained in a textbook that is at once rigorous and practical with many worked examples, exercises with solutions, pseudocode, and sample calculations on images. The introduction lists fast tracks to special topics such as Principal Component Analysis, and ways into and through the book, which abounds with illustrations. The first part describes plane geometry and pattern-generating symmetries, along with some text on 3D rotation and reflection matrices. Subsequent chapters cover vectors, matrices and probability. These are applied to simulation, Bayesian methods, Shannon's Information Theory, compression, filtering and tomography. The book will be suited for course use or for self-study. It will appeal to all those working in biomedical imaging and diagnosis, computer graphics, machine vision, remote sensing, image processing, and information theory and its applications.

DR S. G. HOGGAR is a research fellow and formerly a senior lecturer in mathematics at the University of Glasgow.

MATHEMATICS OF DIGITAL IMAGES

Creation, Compression, Restoration, Recognition

S. G. HOGGAR

University of Glasgow

CAMBRIDGE UNIVERSITY PRESS

CAMBRIDGE UNIVERSITY PRESS
Cambridge, New York, Melbourne, Madrid, Cape Town, Singapore, São Paulo

Cambridge University Press
The Edinburgh Building, Cambridge CB2 8RU, UK
Published in the United States of America by Cambridge University Press, New York

www.cambridge.org
Information on this title: www.cambridge.org/9780521780292

First published in print format 2006

ISBN-13 978-0-511-34846-4 eBook (EBL)
ISBN-10 0-511-34846-0 eBook (EBL)

ISBN-13 978-0-521-78029-2 hardback
ISBN-10 0-521-78029-2 hardback

To my wife, Elisabeth

Contents

Preface

This text is a successor to the 1992 *Mathematics for Computer Graphics*. It retains the original Part I on plane geometry and pattern-generating symmetries, along with much on 3D rotation and reflection matrices. On the other hand, the completely new pages exceed in number the total pages of the older book.

In more detail, topology becomes a reference and is replaced by probability, leading to simulation, priors and Bayesian methods, and the Shannon Information Theory. Also, notably, the Fourier Transform appears in various incarnations, along with Artificial Neural Networks. As the book's title implies, all this is applied to digital images, their processing, compresssion, restoration and recognition.

Wavelets are used too, in compression (as are fractals), and in conjuction with B-splines and subdivision to achieve multiresolution and curve editing at varying scales. We conclude with the Fourier approach to tomography, the medically important reconstruction of an image from lower-dimensional projections.

As before, a high priority is given to examples and illustrations, and there are exercises, which the reader can use if desired, at strategic points in the text; these sometimes form part of the exercises placed at the end of each chapter. Exercises marked with a tick are partly, or more likely fully, solved on the website. Especially after Chapter 6, solutions are the rule, except for implementation exercises. In the latter regard there are a considerable number of pseudocode versions throughout the text, for example ALGO 11.9 of Chapter 11, simulating the d-dimensional Gaussian distribution, or ALGO 16.1, wavelet compression with limited percentage error.

A further priority is to help the reader know, as the story unfolds, where to turn back for justification of present assumptions, and to point judiciously forward for coming applications. For example, the mentioned Gaussian of Chapter 11 needs the theory of positive definite matrices in Chapter 8. In the introduction we suggest some easy ways in, including journeys by picture alone, or by light reading.

Much of the material of this book began as a graduate course in the summer of 1988, for Ph.D. students in computer graphics at the Ohio State University. My thanks are due to Rick Parent for encouraging the idea of such a course. A further part of the book was developed from a course for final year mathematics students at the University of Glasgow.

I thank my department for three months' leave at the Cambridge Newton Institute, and Chris Bishop for organising the special period on Neural Nets, at which I learned so much and imbibed the Bayesian philosophy.

I am indebted to Paul Cockshott for kindly agreeing to be chief checker, and provoking many corrections and clarifications. My thanks too to Jean-Christoph Nebel, Elisabeth Guest and Joy Goodman, for valuable comments on various chapters. For inducting me into Computer Vision I remain grateful to Paul Siebert and the Computer Vision & Graphics Lab. of Glasgow University. Many people at Vision conferences have added to my knowledge and the determination to produce this book. For other valuable discussions at Glasgow I thank Adrian Bowman, Nick Bailey, Rob Irvine, Jim Kay, John Patterson and Mike Titterington.

Mathematica 4 was used for implementations and calculations, supplemented by the downloadable *Image* from the US National Institutes of Health. Additional images were kindly supplied by Lu, Healy & Weaver (Figures 16.35 and 16.36), by Martin Bertram (Figure 17.52), by David Salesin *et al.* (Figures 17.42 and 17.50), by Hughes Hoppe *et al.* (Figures 17.44 and 17.51), and by 'Meow' Porncharoensin (Figure 10.18). I thank the following relatives for allowing me to apply algorithms to their faces: Aukje, Elleke, Tom, Sebastiaan, Joanna and Tante Tini.

On the production side I thank Frances Nex for awesome text editing, and Carol Miller and Wendy Phillips for expertly seeing the book through to publication.

Finally, thanks are due to David Tranah, Science Editor at Cambridge University Press, for his unfailing patience, tact and encouragement till this book was finished.

Introduction

Beauty is in the eye of the beholder ...

Why the quote? Here beauty is a decoded message, a character recognised, a discovered medical condition, a sought-for face. It depends on the desire of the beholder. Given a computer image, beauty is to learn from it or convert it, perhaps to a more accurate original. But we consider creation too.

It is expected that, rather than work through the whole book, readers may wish to browse or to look up particular topics. To this end we give a fairly extended introduction, list of symbols and index. The book is in six interconnected parts (the connections are outlined at the end of the Introduction):

I	*The plane*	Chapters 1–6;
II	*Matrix structures*	Chapters 7–8;
III	*Here's to probability*	Chapters 9–11;
IV	*Information, error and belief*	Chapters 12–13;
V	*Transforming the image*	Chapters 14–16;
VI	*See, edit, reconstruct*	Chapters 17–18.

Easy ways in One aid to taking in information is first to go through following a substructure and let the rest take care of itself (a surprising amount of the rest gets tacked on). To facilitate this, each description of a part is followed by a quick trip through that part, which the reader may care to follow. If it is true that one picture is worth a thousand words then an easy but fruitful way into this book is to browse through selected pictures, and overleaf is a table of possibilities. One might take every second or third entry, for example.

Chapters 1–6 (Part I) The mathematics is geared towards producing patterns automatically by computer, allocating some design decisions to a user. We begin with *isometries* – those transformations of the plane which preserve distance and hence shape, but which may switch left handed objects into right handed ones (such isometries are called *indirect*). In this part of the book we work geometrically, without recourse to matrices. In Chapter 1 we show that isometries fall into two classes: the direct ones are rotations

Context	Figure (etc.)	Context	Figure (etc.)
Symmetry operations	2.16, 2.19	Daubechies wavelets	16.30
Net types	Example 4.20	Fingerprints	16.33
The global scheme	page 75	Wavelets & X-rays	16.35
Face under PCA & K–L	10.15	B-spline designs	17.15, 17.29
Adding noise	11.21	B-spline filter bank	17.32
MAP reconstruction	11.48	Wavelet-edited face	17.39
Martial cats and LZW	12.20	Progressive Venus face	17.52
The DFT	15.2, 15.4	Perceptron	18.15
Edge-detection	15.6, 15.23	Backpropagation	18.20
Removing blur	15.24, 15.27	Kohonen training	18.42
Progressive transmission	15.40	Vector quantisers	18.54
The DCT	15.42, 15.43	Tomography	18.66
Fractals	16.1, 16.17		

or translation, and the indirect ones reflections or glides. In Chapter 2 we derive the rules for combining isometries, and introduce groups, and the dihedral group in particular. In a short Chapter 3 we apply the theory so far to classifying all 1-dimensional or ‘braid’ patterns into seven types (Table 3.1).

From Chapter 4 especially we consider symmetries or ‘symmetry operations’ on a plane pattern. That is, those isometries which send a pattern onto itself, each part going to another with the same size and shape (see Figure 1.3 ff). A plane pattern is one having translation symmetries in two non-parallel directions. Thus examples are wallpaper patterns, floor tilings, carpets, patterned textiles, and the Escher interlocking patterns such as Figure 1.2. We prove the crystallographic restriction, that rotational symmetries of a plane pattern must be multiples of a $1/2$, $1/3$, $1/4$ or $1/6$ turn ($1/5$ is not allowed). We show that plane patterns are made up of parallelogram shaped cells, falling into five types (Figure 4.14).

In Chapter 5 we deduce the existence of 17 pattern types, each with its own set of interacting symmetry operations. In Section 5.8 we include a flow chart for deciding into which type any given pattern fits, plus a fund of test examples. In Chapter 6 we draw some threads together by proving that the 17 proposed categories really are distinct according to a rigorous definition of ‘equivalent’ patterns (Section 6.1), and that every pattern must fall into one of the categories provided it is ‘discrete’ (there is a *lower* limit on how far any of its symmetries can move the pattern).

By this stage we use increasingly the idea that, because the composition of two symmetries is a third, the set of all symmetries of a pattern form a group (the definition is recalled in Section 2.5). In Section 6.3 we consider various kinds of regularity upon which a pattern may be based, via techniques of Coxeter graphs and Wythoff’s construction (they apply in higher dimensions to give polyhedra). Finally, in Section 6.4 we concentrate the theory towards building an algorithm to construct (e.g. by computer) a pattern of any type from a modest user input, based on a smallest replicating unit called a fundamental region.

Chapters 1–6: a quick trip Read the introduction to Chapter 1 then note Theorem 1.18 on what isometries of the plane turn out to be. Note from Theorem 2.1 how they can all be expressed in terms of reflections, and the application of this in Example 2.6 to composing rotations about distinct points. Look through Table 2.2 for anything that surprises you. Theorem 2.12 is vital information and this will become apparent later. Do the exercise before Figure 2.19. Omit Chapter 3 for now.

Read the first four pages of Chapter 4, then pause for the crystallographic restriction (Theorem 4.15). Proceed to Figure 4.14, genesis of the five net types, note Examples 4.20, and try Exercise 4.6 at the end of the chapter yourself. Get the main message of Chapter 5 by using the scheme of Section 5.8 to identify pattern types in Exercises 5 at the end of the chapter (examples with answers are given in Section 5.7). Finish in Chapter 6 by looking through Section 6.4 on 'Creating plane patterns' and recreate the one in Exercise 6.13 (end of the chapter) by finding one fundamental region.

Chapters 7–8 (Part II) After reviewing vectors and geometry in 3-space we introduce n-space and its vector subspaces, with the idea of independence and bases. Now come matrices, representing linear equations and transformations such as rotation. Matrix partition into *blocks* is a powerful tool for calculation in later chapters (8, 10, 15–17). Determinants test row/equation independence and enable n-dimensional integration for probability (Chapter 10).

In Chapter 8 we review complex numbers and eigenvalues/vectors, hence classify distance-preserving transformations (*isometries*) of 3-space, and show how to determine from the matrix of a rotation its axis and angle (Theorem 8.10), and to obtain a normal vector from a reflection matrix (Theorem 8.12). We note that the matrix M of an isometry in any dimension is *orthogonal*, that is $MM^{\mathrm{T}} = I$, or equivalently the rows (or columns) are mutually orthogonal unit vectors. We investigate the *rank* of a matrix – its number of independent rows, or of independent equations represented. Also, importantly, the technique of *elementary row operations*, whereby a matrix is reduced to a special form, or yields its inverse if one exists.

Next comes the theory of quadratic forms $\sum a_{ij}x_ix_j$ defined by a matrix $A = [a_{ij}]$, tying in with eigenvalues and undergirding the later multivariate normal/Gaussian distribution. Properties we derive for matrix norms lead to the *Singular Value Decomposition*: a general $m \times n$ matrix is reducible by orthogonal matrices to a general diagonal form, yielding approximation properties (Theorem 8.53). We include the Moore–Penrose *pseudoinverse* A^+ such that $AX = b$ has best solution $X = A^+b$ if A^{-1} does not exist.

Chapters 7–8: a quick trip Go to Definition 7.1 for the meaning of orthonormal vectors and see how they define an orthogonal matrix in Section 7.2.4. Follow the determinant evaluation in Examples 7.29 then 'Russian' block matrix multiplication in Examples 7.38. For vectors in coordinate geometry, see Example 7.51.

In Section 7.4.1 check that the matrices of rotation and reflection are orthogonal. Following this theme, see how to get the geometry from the matrix in 3D, Example 8.14.

Next see how the matrix row operations introduced in Theorem 8.17 are used for solving equations (Example 8.22) and for inverting a matrix (Example 8.27).

Now look at quadratic forms, their meaning in (8.14), the positive definite case in Table 8.1, and applying the minor test in Example 8.38. Finally, look up the pseudoinverse of Remarks 8.57 for least deviant solutions, and use it for Exercise 24 (end of chapter).

Chapters 9–11 (Part III) We review the basics of probability, defining an *event* E to be a subset of the sample space S of outcomes, and using axioms due to Kolmogorov for probability $P(E)$. After conditional probability, independence and Bayes' Theorem we introduce random variables $X: S \to R_X$, meaning that X allocates to each outcome s some value x in its range R_X (e.g. score x in archery depends on hit position s). An event B is now a subset of the range and X has a pdf (probability distribution function), say $f(x)$, so that the probability of B is given by the integral

$$P(B) = \int_B f(x)\,\mathrm{d}x,$$

or a sum if the range consists of discrete values rather than interval(s). From the idea of average, we define the *expected value* $\mu = E(X) = \int x f(x)\,\mathrm{d}x$ and *variance* $V(X) = E(X - \mu)^2$. We derive properties and applications of distributions entitled binomial, Poisson and others, especially the ubiquitous normal/Gaussian (see Tables 9.9 and 9.10 of Section 9.4.4).

In Chapter 10 we move to random vectors $\boldsymbol{X} = (X_1, \ldots, X_n)$, having in mind message symbols of Part IV, and pixel values. A joint pdf $f(x_1, \ldots, x_n)$ gives probability as an n-dimensional integral, for example

$$P(X < Y) = \int_B f(x, y)\,\mathrm{d}x\,\mathrm{d}y, \quad \text{where } B = \{(x, y): x < y\}.$$

We investigate the pdf of a function of a random vector. In particular $X + Y$, whose pdf is the *convolution product* f^*g of the pdfs f of X and g of Y, given by

$$(f^*g)(z) = \int_{\mathbf{R}} f(t)g(z - t)\,\mathrm{d}t.$$

This gives for example the pdf of a sum of squares of Gaussians via convolution properties of the gamma distribution. Now we use *moments* $E(X_i^r)$ to generate new pdfs from old, to relate known ones, and to prove the *Central Limit Theorem* that $X_1 + \cdots + X_n$ (whatever the pdfs of individual X_i) approaches a Gaussian as n increases, a pointer to the important ubiquity of this distribution.

We proceed to the *correlation* Cov(X, Y) between random variables X, Y, then the covariance matrix Cov$(\boldsymbol{X}) = [\mathrm{Cov}(X_i, X_j)]$ of a random *vector* $\boldsymbol{X} = (X_i)$, which yields a pdf for $\boldsymbol{X}$ if $\boldsymbol{X}$ is multivariate normal, i.e. if the X_i are normal but not necessarily independent (Theorem 10.61). Chapter 10 concludes with *Principal Component Analysis*, or PCA, in which we reduce the dimension of a data set, by transforming to new uncorrelated coordinates ordered by decreasing variance, and dropping as many of the last few variables as have total variance negligible. We exemplify by compressing face image data.

Given a sample, i.e. a sequence of measurements $X_1, \ldots, X_n$ of a random variable X, we seek a statistic $f(X_1, \ldots, X_n)$ to test the hypothesis that X has a certain distribution or, assuming it has, to estimate any parameters (Section 11.1). Next comes a short introduction to the Bayesian approach to squeezing useful information from data by means of an initially vague prior belief, firmed up with successive observations. An important special case is *classification*: is it a tumour, a tank, a certain character, . . . ?

For testing purposes we need *simulation*, producing a sequence of variates whose frequencies mimic a given distribution (Section 11.3). We see how essentially any distribution may be achieved starting from the usual computer-generated uniform distribution on an interval [0, 1]. Example: as suggested by the Central Limit Theorem, the sum of uniform variables $U_1, \ldots, U_{12}$ on [0, 1] is normal to a good approximation.

We introduce Monte Carlo methods, in which a sequence of variates from a suitably chosen distribution yields an approximate n-dimensional integral (typically probability). The method is improved by generating the variates as a *Markov chain* $X_1, X_2, \ldots,$ where X_i depends on the preceding variable but on none earlier. This is called Markov Chain Monte Carlo, or MCMC. It involves finding joint pdfs from a list of conditional ones, for which a powerful tool is a *Bayesian graph,* or *net*.

We proceed to Markov Random Fields, a generalisation of a Markov chain useful for conditioning colour values at a pixel only on values at nearest neighbours. *Simulated annealing* fits here, in which we change a parameter ('heat') following a schedule designed to avoid local minima of an 'energy function' we must minimise. Based on this, we perform Bayesian Image Restoration (Example 11.105).

Chapters 9–11: a quick trip Note the idea of *sample space* by reading Chapter 9 up to Example 9.2(i), then *random variable* in Definition 9.32 and Example 9.35. Take in the binomial case in Section 9.4.1 up to Example 9.63(ii). Now look up the *cdf* at (9.29) and Figure 9.11.

Review *expected value* at Definition 9.50 and the prudent gambler, then *variance* at Section 9.3.6 up to (9.39) and the gambler's return. Now it's time for normal/Gaussian random variables. Read Section 9.4.3 up to Figure 9.20, then follow half each of Examples 9.75 and 9.76. Glance at Example 9.77.

Check out the idea of a joint pdf $f(x, y)$ in Figure 10.1, Equation (10.4) and Example 10.2. Then read up the pdf of $X + Y$ as a convolution product in Section 10.2.2 up to Example 10.18. For the widespread appearance of the normal distribution see the introduction to Section 10.3.3, then the Central Limit Theorem 10.45, exemplified in Figure 10.7. See how the covariance matrix, (10.44), (10.47), gives the n-dimensional normal distribution in Theorem 10.61.

Read the introduction to Chapter 11, then Example 11.6, for a quick view of the hypothesis testing idea. Now the Bayesian approach, Section 11.2.1. Note the meaning of 'prior' and how it's made more accurate by increasing data, in Figure 11.11.

The Central Limit Theorem gives a quick way to simulate the Gaussian/normal: read from Figure 11.21 to 11.22. Then, note how the Choleski matrix decomposition from Chapter 8 enables an easy simulation of the n-dimensional Gaussian.

On to Markov chains, the beginning of Section 11.4 up to Definition 11.52, and their generalisation to Markov random fields, modelling an image, Examples 11.79 and preceding text. Take in Bayesian Image Restoration, Section 11.4.6 above Table 11.13, then straight on to Figure 11.48 at the end.

Chapters 12–13 (Part IV) We present Shannon's solution to the problem of measuring information. In more detail, how can we usefully quantify the information in a message understood as a sequence of symbols X (random variable) from an alphabet $\mathcal{A} = \{s_1, \ldots, s_n\}$, having a pdf $\{p_1, \ldots, p_n\}$. Shannon argued that the mean information per symbol of a message should be defined as the *entropy*

$$H(X) = H(p_1, \ldots, p_n) = \sum -p_i \log p_i$$

for some fixed basis of logarithms, usually taken as 2 so that entropy is measured in bits per symbol. An early vindication is that, if each s_i is encoded as a binary word c_i, the mean bits per symbol in any message cannot be less than H (Theorem 12.8). Is there an encoding scheme that realises H? Using a graphical method Huffman produced the most economical coding that was *prefix-free* (no codeword a continuation of another). This comes close to H, but perhaps the nearest to a perfect solution is an *arithmetic code*, in which the bits per symbol tend to H as message length increases (Theorem 12.35). The idea here extends the method of converting a string of symbols from $\{0, 1, \ldots, 9\}$ to a number between 0 and 1.

In the widely used LZW scheme by Lempel, Ziv and Welch, subsequences of the text are replaced by pointers to them in a dictionary. An ingenious method recreates the dictionary from scratch as decoding proceeds. LZW is used in GIF image encoding, where each pixel value is representable as a byte, hence a symbol.

A non-entropy approach to information was pioneered by Kolmogorov: the information in a structure should be measured as its Minimum Description Length, or MDL, this being more intrinsic than a probabilistic approach. We discuss examples in which the MDL principle is used to build prior knowledge into the description language and to determine the best model for a situation.

Returning to Shannon entropy, we consider protection of information during its transmission, by encoding symbols in a redundant way. Suppose k message symbols average n codeword symbols X, which are received as codeword symbols Y. The *rate* of transmission is then $R = k/n$. We prove Shannon's famous *Channel Coding Theorem*, which says that the transition probabilities $\{p(y|x)\}$ of the channel determine a quantity called the *channel capacity* C, and that, for any rate $R < C$ and probability $\varepsilon > 0$, there is a code with rate R and

$$P(\text{symbol error } Y \neq X) < \varepsilon.$$