

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/315449828>

Foundations of Computer Vision

Book in Intelligent Systems Reference Library · March 2017

DOI: 10.1007/978-3-319-52483-2

CITATIONS

8

READS

5,860

1 author:



James F Peters

University of Manitoba

656 PUBLICATIONS 5,472 CITATIONS

SEE PROFILE

Some of the authors of this publication are also working on these related projects:



Calculus approach to Morphology and its applications [View project](#)



Computer Vision and Image Analysis, Understanding and Processing [View project](#)

Intelligent Systems Reference Library 124

James F. Peters

Foundations of Computer Vision

Computational Geometry, Visual Image
Structures and Object Shape Detection

 Springer

Intelligent Systems Reference Library

Volume 124

Series editors

Janusz Kacprzyk, Polish Academy of Sciences, Warsaw, Poland
e-mail: kacprzyk@ibspan.waw.pl

Lakhmi C. Jain, University of Canberra, Canberra, Australia;
Bournemouth University, UK;
KES International, UK
e-mail: jainlc2002@yahoo.co.uk; jainlakhmi@gmail.com
URL: <http://www.kesinternational.org/organisation.php>

About this Series

The aim of this series is to publish a Reference Library, including novel advances and developments in all aspects of Intelligent Systems in an easily accessible and well structured form. The series includes reference works, handbooks, compendia, textbooks, well-structured monographs, dictionaries, and encyclopedias. It contains well integrated knowledge and current information in the field of Intelligent Systems. The series covers the theory, applications, and design methods of Intelligent Systems. Virtually all disciplines such as engineering, computer science, avionics, business, e-commerce, environment, healthcare, physics and life science are included.

More information about this series at <http://www.springer.com/series/8578>

James F. Peters

Foundations of Computer Vision

Computational Geometry, Visual Image Structures and Object Shape Detection



Springer

James F. Peters
Electrical and Computer Engineering
University of Manitoba
Winnipeg, MB
Canada

ISSN 1868-4394 ISSN 1868-4408 (electronic)
Intelligent Systems Reference Library
ISBN 978-3-319-52481-8 ISBN 978-3-319-52483-2 (eBook)
DOI 10.1007/978-3-319-52483-2

Library of Congress Control Number: 2016963747

© Springer International Publishing AG 2017

This work is subject to copyright. All rights are reserved by the Publisher, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, reuse of illustrations, recitation, broadcasting, reproduction on microfilms or in any other physical way, and transmission or information storage and retrieval, electronic adaptation, computer software, or by similar or dissimilar methodology now known or hereafter developed.

The use of general descriptive names, registered names, trademarks, service marks, etc. in this publication does not imply, even in the absence of a specific statement, that such names are exempt from the relevant protective laws and regulations and therefore free for general use.

The publisher, the authors and the editors are safe to assume that the advice and information in this book are believed to be true and accurate at the date of publication. Neither the publisher nor the authors or the editors give a warranty, express or implied, with respect to the material contained herein or for any errors or omissions that may have been made. The publisher remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Printed on acid-free paper

This Springer imprint is published by Springer Nature
The registered company is Springer International Publishing AG
The registered company address is: Gewerbestrasse 11, 6330 Cham, Switzerland

*This book is dedicated to
Anna Di Concilio, Arturo Tozzi and sweet \mathcal{P}
for the many shapes and pullbacks they
have given to me*

Preface

This book introduces the foundations of computer vision. The principal aim of computer vision (also, called machine vision) is to reconstruct and interpret natural scenes based on the content of images captured by various cameras (see, e.g., R. Szeliski [191]). Computer vision systems include such things as survey satellites, robotic navigation systems, smart scanners, and remote sensing systems. In this study of computer vision, the focus is on *extracting useful information from images* (see, e.g., S. Prince [162]). Computer vision systems typically emulate human visual perception. The hardware of choice in computer vision systems is some form of digital camera, programmed to approximate visual perception. Hence, there are close ties between computer vision, digital image processing, optics, photometry and photonics (see, e.g., E. Stijns and H. Thienpont [188]).

From a computer vision perspective, **photonics** is the science of light in the capture of visual scenes. **Image processing** is the study of digital image formation (e.g., conversion of analogue optical sensor signals to digital signals), manipulation (e.g., image filtering, denoising, cropping), feature extraction (e.g., pixel intensity, gradient orientation, gradient magnitude, edge strength), description (e.g., image edges and texture) and visualization (e.g., pixel intensity histograms). See, e.g., the mathematical frameworks for image processing by B. Jähne [87] and S.G. Hoggar [82], extending to a number of practitioner views of image processing provided, for example, by M. Sonka and V. Hlavac and R. Boyle [186], W. Burger and M.J. Burge [21], R.C. Gonzalez and R.E. Woods [58], R.C. Gonzalez and R.E. Woods and S.L. Eddins [59], V. Hlavac [81], and C. Solomon and T. Breckon [184]. This useful information provides the bedrock for the focal points of computer visionists, namely, image object shapes and patterns that can be detected, analyzed and classified (see, e.g., [142]). In effect, **computer vision** is the study of digital image structures and patterns, which is a layer of image analysis above that of image processing and photonics. Computer vision includes image processing and photonics in its bag of tricks in its pursuit of image geometry and image region patterns.

In addition, it is helpful to cultivate an intelligent systems view of digital images with an eye to discovering hidden patterns such as repetitions of convex enclosures

of image regions and embedded image structures such as clusters of points in image regions of interest. The discovery of such structures is made possible by quantizers. A **quantizer** restricts a set of values (usually continuous) to a discrete value. In its simplest form in computer vision, a quantizer observes a particular target pixel intensity and selects the nearest approximating values in the neighbourhood of the target. The output of a quantizer is called a codebook by A. Gersho and R.M. Gray [55, §5.1, p. 133] (see, also, S. Ramakrishnan, K. Rose and A. Gersho [164]).

In the context of image mesh overlays, the Gersho–Gray quantizer is replaced by geometry-based quantizers. A **geometry-based quantizer** restricts an image region to its shape contour and observes in an image a particular target object shape contour, which is compared with other shape contours that have approximately the same shape as the target. In the foundations of computer vision, geometry-based quantizers observe and compare image regions with approximately the same regions such as mesh maximal nucleus clusters (MNCs) compared with other nucleus clusters. A **maximal nucleus cluster** (MNCs) is a collection of image mesh polygons surrounding a mesh polygon called the nucleus (see, e.g., J.F. Peters and E. İnan on Edelsbrunner nerves in Voronoï tessellations of images [150]). An **image mesh nucleus** is a mesh polygon that is the centre of a collection of adjacent polygons. In effect, every mesh polygon is a nucleus of a cluster of polygons. However, only one or more mesh nuclei are maximal.

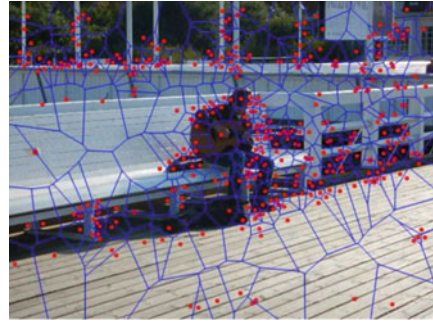
A **maximal image mesh nucleus** is a mesh nucleus with the highest number of adjacent polygons. MNCs are important in computer vision, since what we will call a MNC contour approximates the shape of an underlying image object. A **Voronoï tessellation** of an image is a tiling of the image with polygons. A Voronoï tessellation of an image is also called a Voronoï mesh. A sample tiling of a musician image in Fig. 0.1.1 is shown in Fig. 0.1.2. A sample nucleus of the musician image tiling is shown in Fig. 0.2.1. The red ● dots inside each of the tiling polygons are examples of Voronoï region (polygon) generating points. For more about this, see Sect. 1.22.1. This musician mesh nucleus is the centre of a maximal nucleus cluster shown in Fig. 0.2.2. This is the only MNC in the musician image mesh in Fig. 0.1.2. This MNC is also an example of a Voronoï mesh nerve. The study of image MNCs takes us to the threshold of image geometry and image object shape detection. For more about this, see Sect. 1.22.2.

Each **image tiling polygon** is a convex hull of the interior and vertex pixels. A **convex hull** of a set of image points is the smallest convex set of the set of points. A set of image points A is a **convex set**, provided all of the points on every straight line segment between any two points in the set A is contained in the set. In other words, knowledge discovery is at the heart of computer vision. Both knowledge and understanding of digital images can be used in the design of computer vision systems. In vision system designs, there is a need to understand the composition and structure of digital images as well as the methods used to analyze captured images.

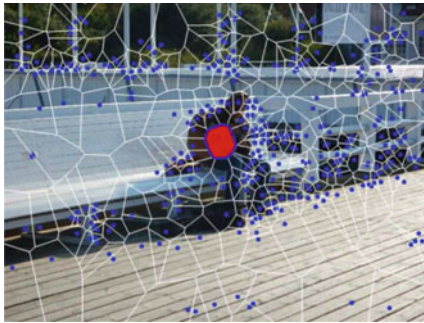
The focus of this volume is on the study of raster images. The sequel to this volume will focus on vector images, which are composed of points (vectors), lines and curves. The basic content of every raster image consists of pixels



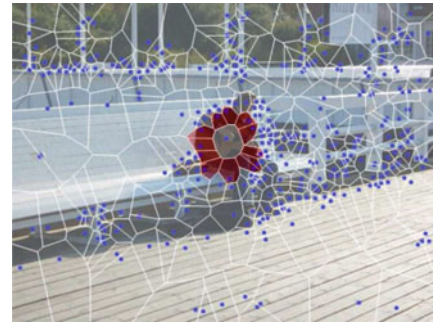
0.1.1: Muscian



0.1.2: Muscian tiling

Fig. 0.1 Voronoi tessellation of musician image

0.2.1: Musician mesh nucleus



0.2.2: Musician maximal nucleus cluster

Fig. 0.2 Maximal nucleus cluster on musician image

(e.g., distinguished pixels called sites or mesh generating points), edges (e.g., common, parallel, intersecting, convex, concave, straight, curved, connected, unconnected), angles (e.g., vector angle, angle between vectors, pixel angle), image geometry (e.g., Voronoi regions [141], Delaunay triangulations [140]), colour, shape, and texture. Many problems in computer vision and scene analysis are solved by finding the most probable values of certain hidden or unobserved image variables and structures (see, e.g., P. Kohli and P.H.S. Torr [96]). Such structures and variables include the topological neighbourhood of a pixel, convex hulls of sets of pixels, nearness (and apartness) of image structures and pixel gradient distributions as well as feature vectors that describe elements of captured scenes.

Other computer vision problems include image matching, feature selection, optimal classifier design, image region measurement, interest point identification, contour grouping, segmentation, registration, matching, recognition, image clustering, pattern clustering in F. Escolono, P. Suau, B. Bonev [45] and in N. Paragios, Y. Chen, O. Faugeras [138], landmark and point shape matching, image warping,

shape gradients [138], false colouring, pixel labelling, edge detection, geometric structure detection, topological neighbourhood detection, object recognition, and image pattern recognition.

In computer vision, the focus is on the detection of the basic geometric structures and object shapes commonly found in digital images. This leads into a study of the basics of image processing and image analysis as well as vector space and computational geometry views of images. The basics of image processing include colour spaces, filtering, edge detection, spatial description and image texture. Digital images are examples of Euclidean spaces (both 2D and 3D). Hence, vector space views of digital images are a natural outcome of their basic character. A **digital image structure** is basically a geometric or a visual topological structure. Examples of image structures are image regions, line segments, generating points (e.g. Lowe keypoints), set of pixels, neighbourhood of a pixel, half spaces, convex sets of pixels and convex hulls of sets of image pixels. For example, such structures can be viewed in terms of image regions nearest selected points or collections of image regions with a specified range of diameters. An **image region** is a set of image points (pixels) in the interior of a digital image. The **diameter** of any image region is the maximum distance between a pair of points in the region). Such structures can also be found in line segments connected between selected points to form triangular regions in 2D and 3D images.

Such structures are also commonly found in 2D and 3D images in the intersection of closed half spaces to form either convex hulls of a set of points or what G.M. Ziegler calls **polytopes** [221]. An **image half space** is the set of all points either above or below a line. In all three cases, we obtain a regional view of digital images. For more about polytopes, see Appendix B.15.

Every image region has a shape. Some region shapes are more interesting than others. The interesting image region shapes are those containing objects of interest. These regional views of images leads to various forms of image segmentations that have practical value when it comes to recognizing objects in images. In addition, detection of image region shapes of interest views lead to the discovery of image patterns that transcend the study of texels in image processing. A **texel** is an image region represented by an array of pixels. For more about shapes, see Appendix B.18 on shape and shape boundaries.

Image analysis focuses on various digital image measurements (e.g., pixel size, pixel adjacency, pixel feature values, pixel neighbourhoods, pixel gradient, closeness of image neighbourhoods). Three standard region-based approaches in image analysis are isodata thresholding (binarizing images), watershed segmentation (computed using a distance map from foreground pixels to background regions), and non-maximum suppression (finding local maxima by suppressing all pixels that are less likely than their surrounding pixels) [212].

In image analysis, object and background pixels are associated with different adjacencies (neighbourhoods) by T. Abera [3]. There are three basic types of neighbourhoods, namely, Rosenfeld adjacency neighbourhoods [171, 102], Hausdorff neighbourhoods [74, 75] and descriptive neighbourhoods in J.F. Peters [142] and in C.J. Henry [77, 76]. Using different geometries, an adjacency

neighbourhood of a pixel is defined by the pixels adjacent to a given pixel. An image **Rosenfeld adjacency neighbourhood** of a pixel p is a set of pixels that are adjacent to p . Adjacency neighbourhoods are commonly used in edge detection in digital images.

A **Hausdorff neighbourhood** of a point p is defined by finding all pixels whose distance from p is less than a positive number r (called the neighbourhood radius). A **descriptive neighbourhood** of a pixel p (denoted by $N(img(x, y), r)$) is the set of pixels with feature vectors that match or are similar to the feature vector that describes $img(x, y)$ (the neighbourhood ‘centre’ of a digital image img) and which are within a prescribed radius r .

Unlike an adjacency neighbourhood, a descriptive neighbourhood can have holes in it, i.e., pixels with feature vectors that do not match the neighbourhood centre and are not part of the neighbourhood. Other types of descriptive neighbourhoods are introduced in [142, Sect. 1.16, pp. 29–34].

The chapters in this book grew out of my notes for an undergraduate class in Computer Vision taught over the past several years. Many topics in this book grew out of my discussions and exchanges with a number of researchers and others, especially, S. Ramanna (those many shapes, especially in crystals), Anna Di Concilio (those proximities, region-free geometry, and seascape shapes like those in Fig. 0.3), Clara Guadagni (those flower nerve structures), Arturo Tozzi (those Borsuk-Ulam Theorem insights and Gibson shapes, Avenarius shapes), Romy Tozzi (remember 8, ∞), Zdzisław Pawlak (those shapes in paintings of the Polish countryside), Lech Polkowski (those mereological, topological and rough set structures), Piotr Artiemjew (those dragonfly wings), Giangiacomo Gerla (those tips (points)–vertices–of UNISA courtyard triangles and spatial regions), Gerald Beer (those moments in Som Naimpally’s life), Giuseppe Di Maio (those insights about proximities), Somashekhar (Som) A. Naimpally (those topological structures), Chris Henry (those colour spaces, colour shape sets), Macek Borkowski (those 3D views of space), Homa Fashandi, Dan Lockery, Irakli Dochviri, Ebubekir İnan (those nearness relations and near groups), Mehmet Ali Öztürk (those beautiful algebraic structures), Mustafa Uçkun, Nick Friesen (those shapes of dwellings), Özlem Umdü, Doungrat Chitcharoen, Çenker Sandoz (those Delaunay triangulations), Surabi Tiwari (those many categories), Kyle Fedoruk (application of computer vision: Subaru EyeSight®), Amir H. Meghdadi, Shabnam Shahfar, Andrew Skowron (those proximities at Banacha), Alexander Yurkin, Marcin Wolski (those sheaves), Piotr Wasilewski, Leon Schilmoeller, Jerzy W. Grzymala-Busse (those insights about rough sets and L^AT_EX hints), Zbigniew Suraj (those many Petri nets), Jarosław Stepaniuk, Witold Pedrycz, Robert Thomas (those shapes of tilings), Marković G. oko (polyforms), Mirosław Pawlak, Pradeepa Yahampath, Gabriel Thomas, Anthony (Tony) Szturm, Sankar K. Pal, Dean McNeill, Giuseppe (Joe) Lo Vetri, Witold Kinsner, Ken Ferens, David Schmidt (set theory), William Hankley (time-based specification), Jack Lange (those chalkboard topological doodlings), Irving Sussman (gold nuggets in theorems and proofs) and Brian Peters (those fleeting glimpses of geometric shapes on the walls).



Fig. 0.3 Seascape shapes along the coastline of Vietri, Italy

A number of our department technologists have been very helpful, especially, Mount-First Ng, Ken Biegun, Guy Jonatschick and Sinisa Janjic.

And many of my students have given important suggestions concerning topics covered in this book, especially, Drew Barclay, Braden Cross, Binglin Li, Randima Hettiarachchi, Enoch A-iyeh, Chidoteremndu (Chido) Chinonyelum Uchime, D. Villar, K. Marcynuk, Muhammad Zubair Ahmad, and Armina Ebrahimi.

Chapter problems have been classified. Those problems that begin with 🚲 are the kind you can run with, and probably will not take much time to solve. Problems that begin with ☕ are the kind you can probably solve in about the time it takes to drink a cup of tea or coffee. The remaining problems will need varying lengths of time to solve.

Winnipeg, Canada

James F. Peters

Contents

1	Basics Leading to Machine Vision	1
1.1	What Is Computer Vision?	2
1.2	Divide and Conquer Approach	3
1.3	Voronoi Diagrams Superimposed on Images	6
1.4	A Brief Look at Computational Geometry	8
1.5	Framework for Digital Images	12
1.6	Digital Visual Space	17
1.7	Creating Your Own Images	18
1.8	Randomly Generated Images	20
1.9	Ways to Display Images	24
1.10	Digital Image Formats	25
1.11	Image Data Types	26
1.12	Colour Images	30
1.12.1	Colour Spaces	30
1.12.2	Colour Channels	30
1.13	Colour Lookup Table	34
1.14	Image Geometry, a First Look	38
1.15	Accessing and Modifying Image Pixel Values	42
1.16	RGB, Greyscale, and Binary (BW) Images	44
1.17	Rosenfeld 8-Neighbourhood of a Pixel	45
1.18	Distances: Euclidean and Taxicab Metrics	50
1.19	False Colours: Pointillist Picture Painting	51
1.19.1	False-Colour an RGB Image Pattern	52
1.19.2	False-Colour a Greyscale Image Pattern	55
1.20	Vector Spaces Over Digital Images	56
1.20.1	Dot Products	57
1.20.2	Image Gradient	58

1.21	What a Camera Sees: Intelligent Systems View	60
1.21.1	Intelligent System Approach in Camera Vision Systems	60
1.21.2	Scene Colour Sensing by Cameras	61
1.22	Image Geometry: Voronoï and Delaunay Meshes on an Image	66
1.22.1	Voronoï Mesh on Car Image	66
1.22.2	What a Voronoï Image Sub-Mesh by Itself Reveals	69
1.23	Nerve Structures	69
1.23.1	Delaunay Mesh on Car Image	75
1.23.2	Combined Voronoï and Delaunay Meshes on Car Image	76
1.24	Video Frame Mesh Overlays	78
1.24.1	Offline Video Frame Processing	79
1.24.2	Real-Time Video Processing	82
2	Working with Pixels	87
2.1	Picture Elements	87
2.2	Separating Colour Image Channels	90
2.3	Colour to Greyscale Conversion	92
2.4	Algebraic Operations on Pixel Intensities	93
2.5	Pixel Selection Illustrated with Edge Pixel Selection	100
2.6	Function-Based Image Pixel Value Changes	109
2.7	Logical Operations on Images	112
2.7.1	Complementing and Logical not of Pixel Intensities	112
2.7.2	Xor Operation on Pairs of Binary Images	113
2.8	Separating Image Foreground From Background	116
2.9	Conjunction of Thresholded Colour Channels	118
2.10	Improving Contrast in an Image	120
2.11	Gamma Transform	122
2.12	Gamma Correction	123
3	Visualising Pixel Intensity Distributions	125
3.1	Histograms and Plots	128
3.1.1	Histogram	129
3.1.2	Stem Plot	131
3.1.3	Plot	131
3.1.4	Surface Plot	131
3.1.5	Wireframe Surface Plot	133
3.1.6	Contour Plot	133
3.2	Isolines	134
3.3	Colour Histograms	137

3.4	Adaptive Thresholding	138
3.5	Contrast Stretching	140
3.6	Histogram Matching	143
4	Linear Filtering	145
4.1	Importance of Image Filtering	145
4.2	Filter Kernels	147
4.3	Linear Filter Experiments	148
4.4	Linear Convolution Filtering	149
4.5	Selecting a Region-of-Interest	151
4.6	Adding Noise to Image	153
4.7	Mean Filtering	155
4.8	Median Filtering	156
4.9	Rank Order Filtering	158
4.10	Normal Distribution Filtering	159
5	Edges, Lines, Corners, Gaussian Kernel and Voronoï Meshes	161
5.1	Linear Function	162
5.2	Edge Detection	164
5.3	Double Precision Laplacian Filter	167
5.4	Enhancing Digital Image Edges	168
5.5	Gaussian Kernel	170
5.6	Gaussian Filter	172
5.7	Gaussian Filter and Image Restoration	174
5.8	Laplace of Gaussian Filter Image Enhancement	175
5.9	Zero-Cross Edge Filter Image Enhancement	176
5.10	Anisotropy Versus Isotropy in Edge Detection	178
5.11	Detecting Edges and Lines in Digital Images	181
5.12	Detecting Image Corners	184
5.13	Image Corner-Based Voronoï Meshes Revisited	187
	5.13.1 Voronoï Tessellation Details	187
	5.13.2 Sites for Voronoï Polygons	188
5.14	Steps to Construct a Corner-Based Voronoï Mesh	189
5.15	Extreme Image Corners in Set of Mesh Generators	190
5.16	Voronoï Mesh on an Image with Extreme Corners	191
5.17	Image Gradient Approach to Isolating Image Edges	194
5.18	Corners, Edges and Voronoï Mesh	196
6	Delaunay Mesh Segmentation	199
6.1	Delaunay Triangulation Generates a Triangular Mesh	200
6.2	Triangle Circumcircles	201
6.3	Constructing a Corner-Based Delaunay Mesh on Image Edges	203

6.4	Centroid-Based Delaunay Image Mesh	204
6.4.1	Finding Image Centroids	205
6.4.2	Finding Image Centroidal Delaunay Mesh.	205
6.4.3	Finding Image Centroidal Voronoï Mesh	206
6.4.4	Finding Image Centroidal Voronoï Superimposed on a Delaunay Mesh	207
7	Video Processing. An Introduction to Real-Time and Offline Video Analysis	211
7.1	Basics of Video Processing	212
7.1.1	Frame Point Processing	213
7.1.2	Image Acquisition	213
7.1.3	Blobs.	214
7.1.4	Frame Tiling and Frame Geometry	215
7.2	Voronoï Tiling of Video Frames	216
7.3	Detection of Shapes in Video Frames.	216
7.4	Measuring Shape Similarity and the Voronoï Visual Hull of an Object	218
7.5	Maximal Nucleus Clusters	220
7.6	Problems.	224
7.7	Shape Distance	232
7.8	Weight Function for Edgelets	234
7.9	Maximum Edgelets.	234
7.9.1	Coarse Contour Edgelets	236
7.9.2	Connected Mesh Regions that are MNCs	237
8	Lowe Keypoints, Maximal Nucleus Clusters, Contours and Shapes.	241
8.1	Image Analysis.	242
8.2	Scene Analysis	244
8.3	Pixel Edge Strength	247
8.4	Cropping and Sparse Representations of Digital Images.	249
8.5	Shape Theory and the Shapes of 2D Image Objects: Towards Image Object Shape Detection	252
8.6	Image Pixel Gradient Orientation and Magnitude	254
8.7	Difference-of-Gaussians	257
8.8	Image Keypoints: D.G. Lowe's SIFT Approach	260
8.9	Application: Keypoint Boundaries of Image Mesh Nuclei	262
8.10	Supra (Outer) Nucleus Contours.	266
8.11	Quality of a MNC Contour Shape	268
8.12	Coarse S2P and S3P (Levels 2 and 3) MNC Contours.	269
8.13	Experimenting with the Number of Keypoints	271

8.14	Coarse Perimeters on Dual MNCs	277
8.15	Rényi Entropy of Image MNC Regions	277
8.16	Problems.	281
9	Postscript. Where Do Shapes Fit into the Computer Vision Landscape?	283
9.1	Shapes in Natural Scenes	284
9.2	Shape Estimates	287
	Appendix A: Matlab and Mathematica Scripts.	291
	Appendix B: Glossary.	361
	References	403
	Author Index.	413
	Subject Index.	417