

Oxford University Computing Laboratory
Wellman Building
Parks Road
Oxford OX1 3QD

Markovian Analysis of Texture: Serial and Parallel Paradigms in Low-Level Vision

by

Christopher Philip Winder

Thesis submitted for the degree of
Doctor of Philosophy
at the
University of Oxford



Programming Research Group
Oxford University Computing Laboratory

St Hugh's College
Michaelmas Term 1991

Technical Monograph PRG-99

ISBN 0-902928-77-5

May 1992

Programming Research Group
Oxford University Computing Laboratory
11 Keble Road
Oxford. OX1 3QD
United Kingdom

Electronic mail: cpw@comlab.ox.ac.uk

Copyright © 1992 C. Philip Winder

Markovian Analysis of Texture: Serial and Parallel Paradigms in Low-Level Vision

by

Christopher Philip Winder

Thesis submitted for the Degree of Doctor of Philosophy
at the University of Oxford

Programming Research Group
Oxford University Computing Laboratory

St Hugh's College
Michaelmas Term 1991

Abstract

Visual texture is a fertile source of information about the physical environment. Texture models should form rich but accessible descriptions of image composition and appearance. Markovian representations make explicit the variability of natural textures, but manipulation of current models is computationally demanding. This practical limitation enforces approximations and use of the simplified auto-normal form.

We propose two novel frameworks for Markovian texture analysis, and illustrate and quantify their advantages by adopting Bayesian classification of 33 Brodatz textures as a benchmark.

1. Simple spatially-parallel image filtering is computationally attractive, but our experiments demonstrate the limitations of segmentation algorithms responding to first-order differences of Gabor amplitude. We harness the efficiency of Gabor filtering within a hybrid *Gabor-Markov* framework by describing arrangements of local image features with random field models.
2. Our experimental appraisal of Gabor-Markov models established the importance of pre-processing image data prior to statistical analysis. We propose two families of *Sampled-Markov* models employing concise representations derived directly from the image data.

Both paradigms are more efficient and robust than a conventional Markovian analysis. Without reducing classifier accuracy, computational load was decreased by 88% using Gabor-Markov, and by 96% using Sampled-Markov models. Despite the improvements achieved by Gabor-Markov models, Smooth-Sampled Markov models perform better and have achieved 100% accuracy in our tests. We explain their superior performance by showing a strong correlation between classification accuracy and fidelity of the statistical modelling.

Acknowledgements

I am heavily indebted to my supervisor Andrew Blake for accepting me as his charge, for providing the inspiration behind this work, and for keeping me on the "straight and narrow". Without his patience, persistence and guidance during the past 18 months, this thesis would not have been possible.

I am also grateful to Ian Page for many helpful and enthusiastic discussions, and for supervising me during part of my time at Oxford. Thanks are due to Mike Brady, Bernard Sufrin and Lionel Tarassenko for their constructive advice and encouragement. I thank David Sinclair, Wayne Luk and Phil Richards, as well as Andrew, Ian and Bernard for many helpful comments on earlier drafts. Enforced brevity prevents me from acknowledging individually the many other people who have contributed to my progress.

This work was conducted at Oxford University Computing Laboratory and supported by a grant from the U.K. Science and Engineering Research Council. I gratefully acknowledge a scholarship awarded by St Hugh's College and funded by the I.B.M. U.K. Scientific Centre, Winchester. I am also indebted to my parents (morally and financially) for their patient support and understanding.

Finally, I would like to record my deep gratitude to Alison for holding my hand during the worst moments and for her unending love and support. This thesis has benefited considerably from her thoughtful comments and proof-reading. Little did she think married life would be like this!

Oxford
January, 1992

Contents

Abstract	v
Acknowledgements	vii
Contents	ix
List of Figures	xi
1 Introduction	1
1.1 Visual Texture	2
1.2 Analysis of Visual Texture	3
1.3 Markovian Analysis of Texture	6
1.4 Thesis Overview	8
2 Models of Visual Attention: Serial and Parallel Paradigms	13
2.1 Low-Level Human Visual Processing	14
2.2 Models of Pre-Attentive Human Vision	16
2.3 Characteristics of Human Texture Primitives	22
2.4 Origins of Dual Paradigm Processing	28
3 Approaches to the Representation and Analysis of Visual Texture	33
3.1 Introduction: Texture Representations	34
3.2 Statistical Approaches to Texture Analysis	37
3.3 Structural Approaches to Texture Analysis	44
3.4 Texture Classification Algorithms	46
3.5 Image Segmentation Algorithms	49
3.6 Discussion of Current Approaches	52
4 Gabor Segmentation Experiments and Feature Extraction Algorithms	55
4.1 Theoretical Properties of Gabor Filters	56
4.2 Relation to Other Texture Measures	64
4.3 Applications of Gabor Filtering: A Review	65
4.4 Gabor Segmentation Experiments	69
4.5 Gabor Signatures: Parameterised Image Models	81
4.6 Feature Estimation using Response Moments	87
4.7 Conclusions	94

5	Texture Analysis with Gibbs–Markov Random Field Models	101
5.1	Foundations of Gibbs–Markov Random Fields	102
5.2	Estimation of Parameters for Gibbs–Markov Models	116
5.3	Texture Synthesis from Gibbs–Markov Models	121
5.4	Gibbs–Markov Texture Classification	125
5.5	Restoration of Degraded Images	126
5.6	Segmentation of Textured Images	128
5.7	Motivation for Gabor–Markov Framework	131
6	Markovian Texture Classification: A Comparison of Novel and Conventional Models	135
6.1	Choice of Auto-Normal MRF Model	136
6.2	Pixel Images: Conventional Analysis	137
6.3	Feature Images: Gabor–Markov Models	141
6.4	Implementation of the Classifier	146
6.5	Performance of Texture Classifiers	151
6.6	Performance of Sampled-Markov Classifiers	157
6.7	Comparison of Texture Models	163
7	Consequences of Adopting Auto-Normal Models	175
7.1	Relation between Classifier Accuracy and Fidelity of our Statistical Model	176
7.2	Comments on the Influence of Image Blur	194
7.3	Comments on the Influence of Additive Noise	201
7.4	Comments on the Influence of Image Sampling	205
7.5	Conclusions: Modelling Accuracy Determines Classifier Performance . .	212
8	Conclusions, and Suggestions for Further Work	215
8.1	Markovian Analysis of Texture: Conclusions	216
8.2	Improved Gabor Feature Representations	219
8.3	Improved MRF Models	222
8.4	Applications to Image Analysis	227
	Bibliography	231
	Glossary of Principal Notation	241
A	Appendices	247
A.1	Parallel Vision Architectures	247
A.2	Properties of Gabor Filters	251
A.3	Alternative Gabor Feature Estimators	259
A.4	Marginal Densities of Proposed Gabor–Markov Models	266
A.5	Properties of Modified Textures	269

List of Figures

1.1	Brodatz texture montages	2
1.2	Intensity edge segments detected in Brodatz montages	4
1.3	Texture boundaries detected in Brodatz images by Gabor filtering	5
1.4	Synthetic Brodatz textures derived from Markov models	7
1.5	Gabor features extracted from Brodatz montages	9
1.6	Our Brodatz image set	11
2.1	Schematic attentive and pre-attentive response curves	16
2.2	Search elements with similar textons	18
2.3	Distinction between implicit and explicit grouping	19
2.4	Schematic diagram of feature-integration theory	20
2.5	Artificial displays demonstrating search asymmetry	21
2.6	Artificial T-L search displays	22
2.7	Effect of anisotropic stretching on texton density	24
2.8	Artificial display used in combined counting-identification task	26
2.9	Artificial image with two levels of structure	27
2.10	Formation of perceptual groups by spatial filtering	28
4.1	Profiles of a typical Gabor kernel	58
4.2	Fourier transform of a typical Gabor kernel	59
4.3	Information diagram for Gabor and wavelet image representations	63
4.4	Zebra image and power spectrum	68
4.5	Artificial "Segregation" and "Pop-out" images	71
4.6	Raw and demodulated Gabor filter output from "Pop-out" image	71
4.7	"Segregation" image after modal filtering	72
4.8	"Pop-out" image after modal filtering	72
4.9	Effect of contrast operator on filtered "Pop-out" image	73
4.10	Effect of gradient operator on filtered "Pop-out" image	74
4.11	Phase effects with Gabor filters	76
4.12	Artificial images after filtering with a local kernel	76
4.13	Denim image after modal filtering	77
4.14	Successful Gabor energy segmentations of natural textures	78
4.15	Unsuccessful Gabor energy segmentations of natural textures	80
4.16	Gabor signature of untranslated LINE primitive	84
4.17	Gabor signature of translated LINE primitive	84

4.18	Response surfaces for the LINE primitive	86
4.19	Gabor signature of the SINU primitive	86
4.20	Gabor features extracted from herringbone-canvas montage	92
4.21	Gabor features extracted from "Pop-out" image	93
4.22	Gabor features extracted from zebra image	95
4.23	Gabor features extracted from herringbone image	95
4.24	Gabor features extracted from Westminster image	96
4.25	Gabor features extracted from mat image	97
4.26	Gabor features extracted from Brodatz images	98
5.1	Markov neighbourhood structure	107
5.2	Gibbs random field clique types	110
6.1	Shape of true-likelihood function near maximum	140
6.2	Synthetic Brodatz image set	142
6.3	Comparison of Gabor-Markov angular field models	144
6.4	Classification procedure	150
6.5	Effect of noise and blur on Brodatz image	154
6.6	Scatter matrix of pixel and composite-feature classifiers	156
6.7	Sub-sampled Brodatz image	157
6.8	Comparison of original and sampled Markov neighbourhoods	158
6.9	Smooth-sampled Brodatz image	161
6.10	Accuracy of unsampled pixel classifier	165
6.11	Accuracy of unsampled pseudo-likelihood pixel classifier	165
6.12	Accuracy of Gabor-Markov orthogonal-normal orientation classifier	166
6.13	Accuracy of Gabor-Markov Resultant composite-feature classifier (<i>versus</i> sampling period)	166
6.14	Accuracy of Gabor-Markov Resultant composite-feature classifier (<i>versus</i> MRF order)	167
6.15	Accuracy of Gabor-Markov Profile composite-feature classifier	167
6.16	Accuracy of unsampled pixel classifier with noise	168
6.17	Accuracy of Gabor-Markov Resultant composite-feature classifier with noise	168
6.18	Accuracy of unsampled pixel classifier with blur	169
6.19	Accuracy of Gabor-Markov feature classifiers with blur	169
6.20	Accuracy of Sub-Sampled Markov classifier	170
6.21	Accuracy of Sub-Sampled Markov classifier with noise	170
6.22	Accuracy of Sub-Sampled Markov classifier with blur	171
6.23	Accuracy of Smooth-Sampled Markov classifier	171
6.24	Accuracy of Smooth-Sampled Markov classifier with noise	172
6.25	Accuracy of Smooth-Sampled Markov classifier with blur	172
6.26	Comparative accuracy of Gabor-Markov, Sampled-Markov and conventional classifiers with noise	173
6.27	Comparative accuracy of Gabor-Markov, Sampled-Markov and conventional classifiers with blur	173

7.1	Agreement with auto-normal model of unsampled Brodatz images . . .	178
7.2	Agreement with auto-normal model of synthetic and mis-matched textures	178
7.3	Influence of sampling period and MRF order on variance of pixel models	180
7.4	Agreement with auto-normal model of perturbed synthetic textures . .	181
7.5	Agreement with auto-normal model of corrupted synthetic images . . .	184
7.6	Agreement with auto-normal model of sub-sampled Brodatz images . .	187
7.7	Agreement with auto-normal model of smooth-sampled Brodatz images	187
7.8	Agreement with auto-normal model of sampled corrupted synthetic textures	187
7.9	Agreement between auto-normal and Gabor-Markov models	188
7.10	Agreement between auto-normal and sampled Gabor-Markov models . .	189
7.11	Agreement between auto-normal and sampled Gabor-Markov models with corruption	191
7.12	Illustration of improved Gabor-Markov models	192
7.13	Agreement between auto-normal and improved Gabor-Markov models .	193
7.14	Scatter matrices for uncorrupted pixel classifiers	198
7.15	Scatter matrices for pixel classifiers with blur	198
7.16	Histograms of MRF variance for unsampled pixel models	199
7.17	Scatter matrices for pixel classifiers with noise	203
7.18	Illustration of frequency aliasing arising from sampling	206
7.19	Power spectra of dense and sampled Brodatz images	207
7.20	Effect of sampling on contrast of spectral peak	208
7.21	Model power spectra of dense and sampled Brodatz images	209
7.22	Scatter matrices for Sampled-Markov classifiers with corruption	210
7.23	Power spectra of dense and sampled Gabor-Markov features	213
8.1	Examples of proposed Gabor "phase" feature	221
8.2	Anti-aliased Brodatz image	224
8.3	Effect of anti-aliasing on image spectra and correspondence with auto-normal model	225
8.4	Modified Markov neighbourhood structure	226
8.5	Gabor features suitable for modelling with Gibbs random fields	229
A.1	DisPuter Architecture	250
A.2	Gabor orientation and frequency bandwidths	252
A.3	Relation between Gabor aliasing error and sampling density	254
A.4	Filter kernel used to ensure that local image mean is zero	255
A.5	Gabor energy variation for LINE primitive	262
A.6	Gabor response variation for SINU primitive	264
A.7	Marginal density for orthogonal-normal model	268
A.8	Effect of temporal sampling on the frequency spectrum	272
A.9	Spectra of blur and window functions	273

1

Introduction

Visual texture provides a fertile source of information about the nature of physical objects and surfaces, but is not exploited fully by current machine vision systems. Two main obstacles have retarded progress in this area: the failure of texture models to provide rich but accessible descriptions of image structure; and the prohibitively high computational requirements of many algorithms operating on these representations. One source of modelling difficulty is the ill-defined and variable nature of real textures, whose hierarchical organisation may encompass many levels of structure, for which a traditional analysis is not appropriate. Probabilistic representations using Markov random field models acknowledge stochastic character and the influence of spatial context, but are unwieldy and insufficiently abstract.

Our approach is to develop novel representations of texture retaining the descriptive power of conventional Markov models but with increased computational efficiency, robustness, and level of abstraction. We propose a hybrid *Gabor-Markov* framework for texture analysis, drawing on the attractive theoretical and practical properties of Gabor filters. We also propose two families of *Sampled-Markov* models, employing concise representations derived directly from the image data. Bayesian texture classification benchmarks demonstrate that both these paradigms out-perform conventional Markovian approaches, meeting our stated objectives. For the same accuracy as a conventional Markov classifier, computational load was decreased by 88% using our Gabor-Markov composite-feature model, and by 96% using Smooth-Sampled Markov models. Despite the enhancements achieved by Gabor-Markov models, Smooth-Sampled Markov models are preferred, and have achieved 100% accuracy in our trials. Accordingly, we suggest that a conventional Markovian analysis should be retained only for very small image regions, which Gabor-Markov or Sampled-Markov models do not describe reliably.

We introduce the concept of visual texture and the basis for current approaches in

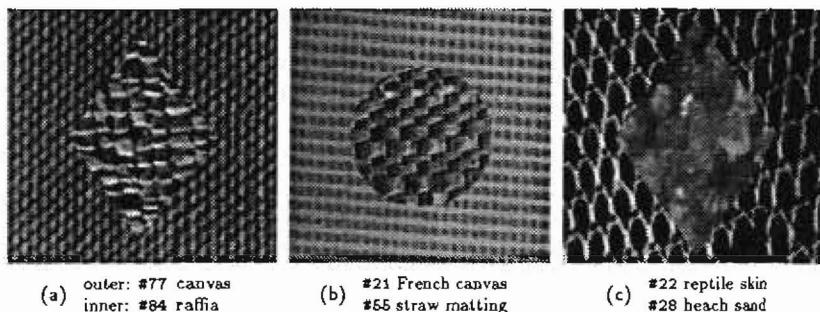


FIGURE 1.1: **Brodatz texture montages.** Images formed by overlaying portions of Brodatz textures (Figure 1.6), the boundary forming a diamond or a circle.

this Chapter, and briefly summarise the layout of this thesis. Reviews of models of human visual processing (Chapter 2) and of previous approaches to texture analysis (Chapter 3) further motivate our approach.

1.1 Visual Texture

Visual texture is concerned with spatial patterns and arrangements, and is a fundamental property of physical objects. Intuitively, “texture” refers to local image context, describing relationships between nearby spatial components, although no formal definition has proved entirely adequate. Models of visual texture accordingly contain two components: primitive elements, or *microtexture*; and their spatial arrangement, or *macrotexture*. Single pixels¹ form the simplest microtextures, but our concept of texture may be expanded recursively, with macrotexture at one level forming microtexture at the next, to represent complex structured spatial arrangements. Natural textures are often observed to have a hierarchical organisation of this type, possessing quite different characteristics at different resolutions (Ahuja & Rosenfeld, 1981; Marr, 1976).

Three examples of images containing visual texture are shown in Figure 1.1, formed by combining portions of textures from the celebrated Brodatz album (Brodatz, 1966). These particular textures are fairly regular, and suitable divisions into micro and macrotexture may seem obvious. Note, however, that straw loops in matting (#55)² or raffia (#84) may be seen as textures in their own right, and hence form a hierarchy. This point is further illustrated by herringbone weave (#17; Figure 1.6): the material contains bands arising from different orientations of the weave, but each band also forms a valid image texture. Another facet of texture is illustrated by beach sand (#28) or by clouds (#90): these textures are amorphous, and do not decompose readily into distinct “micro” and “macro” components. Nonetheless, all these images, and the remainder of

¹A pixel is a single element in an image array, whose value represents intensity.

²The notation #56 refers to the plate number of a Brodatz texture shown in Figure 1.6 (page 11).

our Brodatz set (Figure 1.6), are valid examples of the phenomenon of spatial variation called “texture”. General-purpose texture analysers seek to manipulate images of this type — recognising, classifying, segmenting, locating, and restoring them — in a manner that is computationally efficient but robust to image corruption and noise.

1.2 Analysis of Visual Texture

Vision is the most powerful of our five senses, allowing us to perceive the world about us: to observe meaningful physical objects, and deduce their spatial relations; to track sequences of events, and make fine discriminations. Visual texture lies behind many of these abilities, and camouflage reduces the completeness with which we perceive our environment. In view of the subjective ease with which many tasks may be facilitated “just by looking”, it might have been tempting to assume that machine vision systems would soon enjoy capabilities quantitatively similar to our own, but this has not proved to be the case. Early wholistic or “Gestalt” approaches made little headway, and it has proved more profitable to examine separate visual modules in isolation (stereopsis, motion, colour, *etc.*), each processing different aspects of the image data without task-specific knowledge (Brady, 1981). Together, these modules form low-level vision.

1.2.1 Links with Human Vision

Some complex visual tasks appear very straightforward to us — discriminating the textures shown in Figure 1.1, or recognising our grand-mother in a crowd — but other visual judgements seem more difficult, and require detailed scrutiny. Controlled experiments confirm the presence of both “serial” and “parallel” processes in low-level human vision, although the dichotomy is less distinct than the conventional “attentive” and “pre-attentive” labels for these paradigms may suggest (Treisman & Sato, 1990). Current psychological models propose that an early pre-attentive process first locates potentially “significant” image locations by performing rudimentary filtering, and that these are subsequently examined attentively in greater detail, selectively and sequentially (Wolfe & Cave, 1990). Only the efficiency of pre-attentive guidance allows the illusion of immediate and complete perception to be sustained.

Unfortunately, psychological models are often expressed in ambiguous verbal terms, and cannot be subjected to computational analysis or exploited directly by machine vision systems (Broadbeat, 1987; Winder, 1988b). Prompted by these ideas, however, analysis of the complexity of visual search has suggested that *complete* analysis of the image may be infeasible, and hence that *selective* interpretation is required (Gurari & Wechsler, 1982; Tsotsos, 1987). Heuristic assumptions about the nature of the physical environment may guide this choice (Marr, 1982).

1.2.2 Structural and Statistical Approaches

Texture analysis has a dual purpose: to locate boundaries; and to characterise regions. Assumptions of spatial and boundary coherence suggest that image texture provides

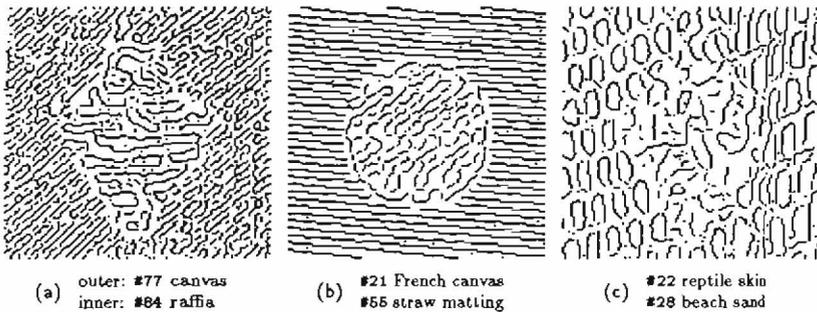


FIGURE 1.2: Intensity edge segments detected in Brodatz montages. Edge segments found by the Canny (1983) edge-detector in our Brodatz montages (Figure 1.1). Although these edges correctly reflect microtexture structure, they have failed to respond to the border between different texture types. The Canny scale parameter was 2 pixels, and hysteresis threshold ratio 6:2.

a useful indication of the properties of physical objects present in a scene, which are of ultimate interest (Marr, 1982). Historically, discontinuities in image structure have been modelled by one-dimensional changes in *first-order* properties, and located by “edge-detectors”. First-order image properties are strongly influenced by the image-acquisition environment, by variation in lighting or camera parameters, in addition to intrinsic surface texture, and hence are unreliable indicators of physical structure. Higher-order properties defined over an image neighbourhood, or “texture”, form a more robust guide to physical surfaces and their discontinuities. Approaches to visual texture are commonly divided into “structural” and “statistical” models, and this separation is helpful in capturing the major aspects of texture analysis, but there is no rigid distinction between them (Haralick, 1979).

Analysis of Texture Structure

Canny’s (1983) “optimal” intensity edge-detector enjoys high popularity and performs well for noisy images, but edge segments detected in our Brodatz montages illustrate some drawbacks of a first-order approach to texture analysis (Figure 1.2). Although these edges form a reasonable description of local image structure, they do not define a clear texture boundary because differences in local intensity reflect only *microtexture*. When different image textures have the same mean intensity, their *macrotexture* boundary cannot be detected directly by this method. In order to segment images into different textured regions, it is first necessary to construct *macrotexture* models by piecing together local edge segments. While this approach is feasible for regular textures of the sort we have illustrated (Figure 1.1a, b and #22), if time-consuming and error-prone, it breaks down for more amorphous textures of the type to which human vision adapts so easily (e.g. #28, #90; Figure 1.6).

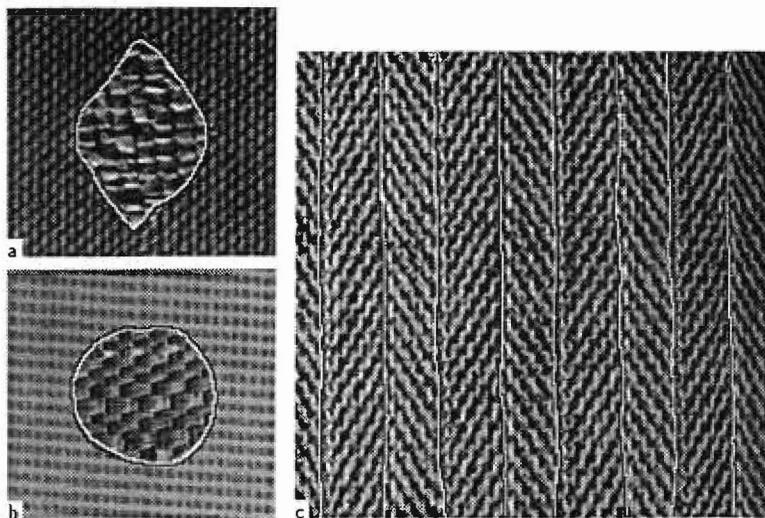


FIGURE 1.3: Texture boundaries detected in Brodatz images by Gabor filtering. Each image was filtered with two tuned Gabor kernels, and blurred with a space-constant of five pixels. The superimposed border marks locations where the dominant response changes. (a) cotton canvas-rafia montage, 128×128 pixels (Figure 1.1a); (b) French canvas-straw matting montage, 128×128 pixels (Figure 1.1b); (c) herringbone image, 256×256 pixels (#17; Figure 1.6). Gabor edges reflect texture boundaries rather than microtexture detail (compare Figure 1.2). This simple method was *not* successful with our reptile-sand montage (Figure 1.1c).

Analysis of Texture Statistics

An alternative “statistical” approach³ to texture is to create more complex image models, and hence deduce filters responding directly to *textured* rather than intensity edges. Textures are characterised indirectly by simple derived properties, or local “features”, with the expectation that distinct textures possess different features, leading to first-order differences in the filtered image. Texture processing requires analysis of a spatially-varying signal coupled with accurate boundary localisation: Gabor filtering offers the best compromise between these conflicting requirements (Gabor, 1946). Acceptable segmentations of two of our Brodatz montages (Figure 1.1a,b) and herringbone weave (#17) were achieved with only light computational effort by detecting first-order differences in Gabor filter output (Figure 1.3). Algorithms responding to differences in Gabor filter output offer capabilities qualitatively similar to pre-attentive human vision, but failed to locate an acceptable border with our reptile-sand montage (Figure 1.1c; data not shown). Further examples are presented in Chapter 4 (e.g. Figure 4.14 on page 78).

³In this context, “statistic” refers to a derived property or feature, and does not necessarily imply stochastic modelling.

A second difficulty for statistical approaches is that an inappropriate choice of features may lead to many false boundaries occurring *within* a texture of a single type since spatial variation is an intrinsic characteristic. These inherent limitations imply that statistical approaches to texture analysis are not completely reliable, and may be considered analogous to pre-attentive human vision. In this role, they form an initial rudimentary image representation, marking approximate texture boundaries and regions suitable for further processing.

1.3 Markovian Analysis of Texture

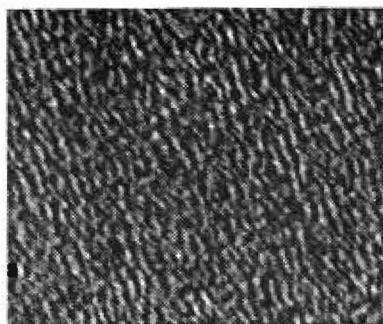
Structural models describe texture by regular tessellations of the image plane, coloured by an alphabet of microtexture primitives; statistical models assume that derived texture "features" are spatially homogeneous. Neither of these approaches makes explicit a fundamental property of real textures: *variability*. Extraneous variation arises from distortions and noise introduced by the image acquisition system, and cannot be fully eliminated; but the chief source of image variety arises from the inherently variable nature of real textures. Notions of variability, optimality, and hypothesis-testing are central to statistical-stochastic study,⁴ and hence this forms a natural framework within which these effects may be modelled. This route is available to both "structural" and "statistical" descriptions, but much of the literature has concentrated on the subject of statistical pattern recognition (Devijver & Kittler, 1982).

Operating within a statistical-stochastic framework, Markov random field models address an important weakness by making explicit the influence of spatial context. Textures are characterised by the marginal probability density of each pixel conditioned on its image neighbourhood, modifying pixel behaviour according to image context. A statistical "figure of merit" for the complete texture is formed by computing the joint likelihood of all image pixels according to the Markov model, allowing competing hypotheses to be compared. The joint distribution may be sampled to form *synthetic* textures (Figure 1.4), but note that synthesis is a *stochastic* and hence unrepeatable process, and that no valid point-to-point comparisons with the original textures may be made. Visual comparison of synthetic textures with the original is nonetheless an attractive method for assessing the quality of the model. Pressed-cork (#4), French canvas (#21) and cotton canvas (#77; Figure 1.6) are homogeneous textures, and are visually similar to their synthetic counterparts (Figure 1.4a-c), but the reverse is true for reptile skin (#28), straw matting (#55), and raffia (#84), which all display a higher level of structure (Figure 1.4d-f). Markov models operate successfully when there is no clear distinction between micro and macrotecture, and can discriminate almost all the textures in our Brodatz set.⁵ They have been successfully applied to a variety of classic image-processing problems, including classification, segmentation and restoration.

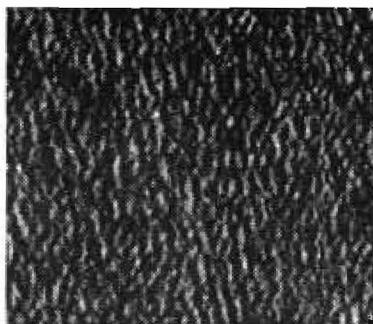
One deficiency of Markov representations is apparent in Figure 1.4d-f: syntactic structure or texture grammar is treated poorly, despite the influence of spatial context, because the same stochastic model applies to all image locations. This weakness is

⁴By "statistical-stochastic", we do mean to imply modelling of random behaviour.

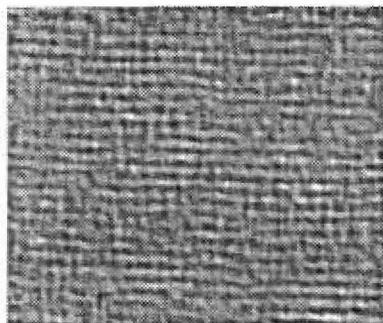
⁵Conventional Markov models classify image quarters with an error rate of 6%, falling to less than 1% after the modifications proposed in this thesis.



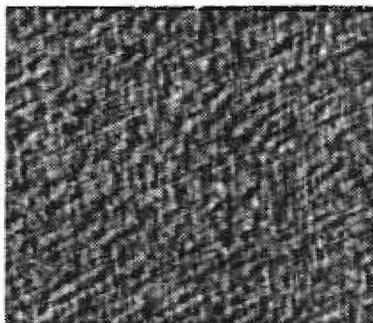
(a) #4 pressed cork



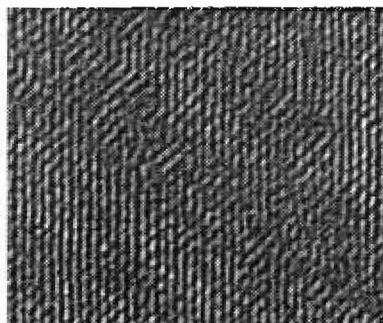
(d) #22 reptile skin



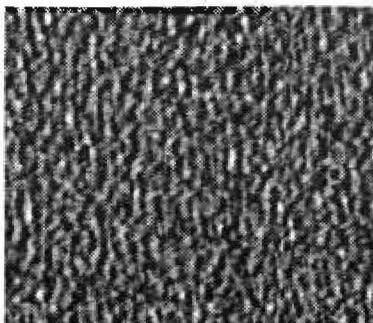
(b) #21 French canvas



(e) #56 straw matting



(c) #77 cotton canvas



(f) #84 raffia

FIGURE 1.4: Synthetic Brodatz textures derived from Markov models. Synthetic images formed from fifth-order true-likelihood Markov random field models of natural Brodatz textures (Figure 1.6), 192×168 pixels. (a–c) visual similarity is strong when the original texture is homogeneous; (d–f) these models have failed to capture the structure of the original textures.

partially allayed by *hierarchical* random field models, which form a richer description of image structure. Conventional Markov models operate at a very low level, describing the spatial arrangement of individual pixels, and this lack of abstraction makes them sensitive to imaging parameters such as rotation and magnification. Exhaustive evaluation of all possible hypotheses is feasible during texture classification because the number of texture classes is not excessively large. This optimal approach is not practical during image segmentation because the number of possible scene partitions rises exponentially with image size (Tsotsos, 1987). Heuristic assumptions or approximations ease the computational burden, but these detract from the “optimal” nature of the statistical framework. High computational appetite is a factor common to all Markov texture algorithms, particularly for hierarchical models, and forms a significant restriction on their wider application for image analysis.

1.4 Thesis Overview

The chief objective of this thesis is to propose novel texture models combining computational efficiency with the optimal statistical framework and superior descriptive power offered by Markov random fields. Our approach is to modify conventional Markov models to:

- raise their computational efficiency;
- improve their robustness with respect to image noise and blur;
- increase their level of abstraction, so they are less sensitive to imaging parameters.

Chapters 2, 3 and 5 consist mainly of review and analysis of relevant literature and theoretical foundations. Most of the novelty of our contribution lies in Chapters 4, 6 and 7, with suggestions for further research in Chapter 8. In the remainder of this Chapter, we describe our thesis in greater detail.

Models of human vision have provided many insights into the nature of visual information processing, at both theoretical and practical levels. Psychologists report that the major feature of low-level human vision is an apparent dichotomy between “attentive” and “pre-attentive” processing, but the precise computational form of these paradigms has so far eluded empirical research (Chapter 2). A similar division is often drawn for texture analysis in machine vision, into “structural” and “statistical” approaches, and we review the main characteristics of current techniques (Chapter 3). The hierarchical nature of visual texture is made explicit by structural approaches, which seek to form separate descriptions for micro and macro-structure, and are most often used in conjunction with stochastic models. Statistical approaches decompose the image into local derived properties or “features”, whose selection is often motivated by models of pre-attentive human vision. Local image features may often be computed very efficiently, but cannot match the richness of structural representations.

Texture analysis is ultimately concerned with observing physical objects, and hence demands simultaneous measurement of spatial and spectral image properties. Optimal compromises of these conflicting requirements are achieved uniquely by Gabor kernels,

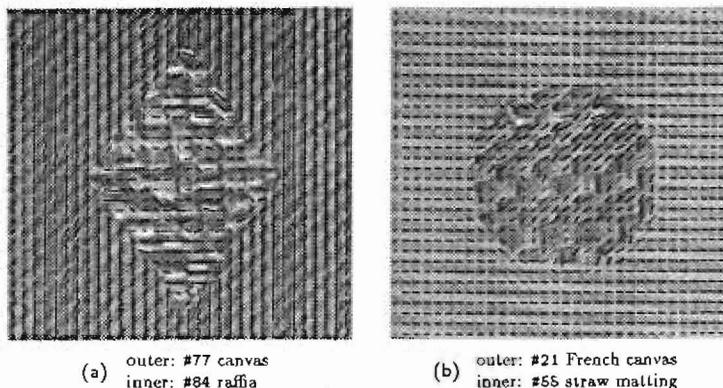


FIGURE 1.5: **Gabor features extracted from Brodatz montages.** Feature vectors were estimated from Brodatz montages (Figure 1.1 *a, b*) using our Resultant method (§4.6), and sampled every four pixels. Needle intensity represents “error”, and length is proportional to “contrast”.

which have recently found application in low-level vision. Our experiments with natural and synthetic textures demonstrate the potential of Gabor filtering for image analysis, and confirm that simple efficient algorithms responding to first-order differences in filter output segment suitably-constrained images adequately (Chapter 4). Limitations of this approach arise from over-simplified texture models, which assume that textures contain few significant spectral components, and residual spatial variability *within* each filtered region interferes with attempts to extract texture boundaries.

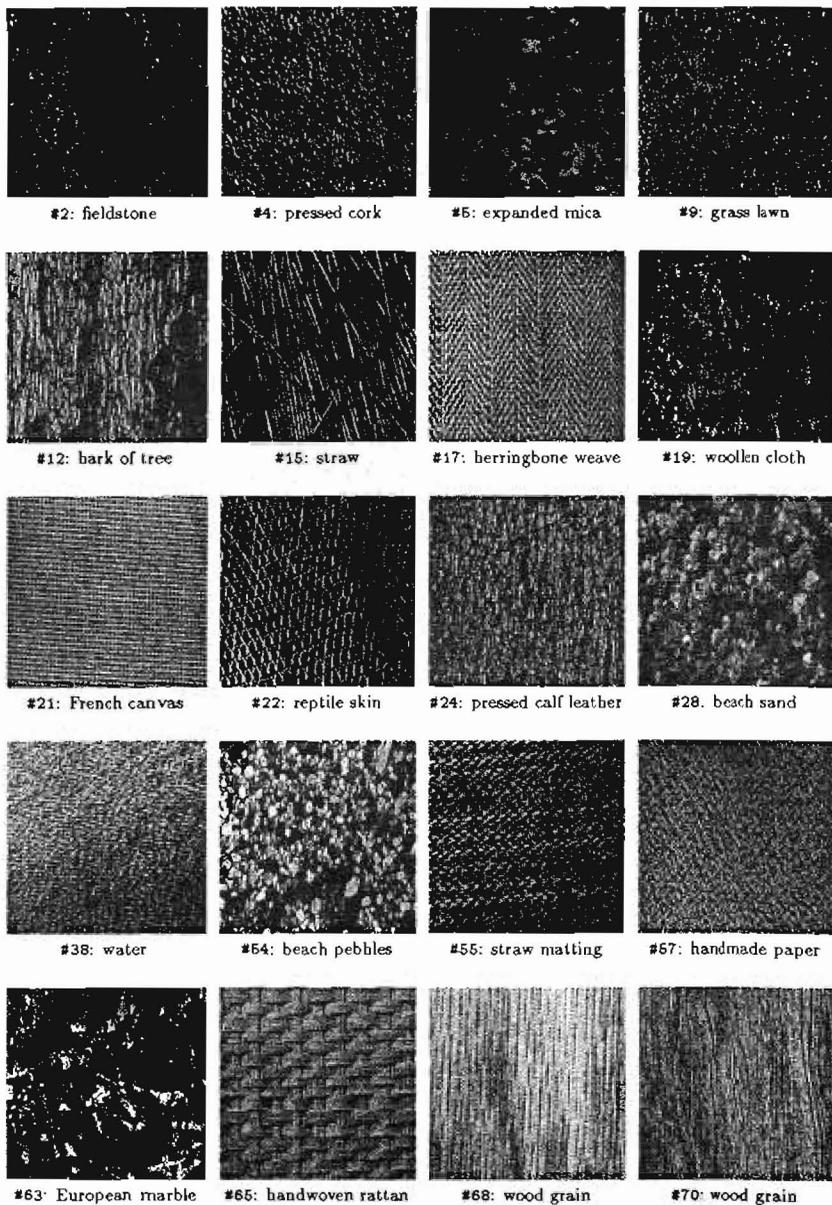
We propose to *exploit* rather than suppress fluctuations in filter output, describing the spatial arrangement of Gabor “features” with Markov random fields to form hierarchical *Gabor-Markov* models. The role of our Gabor-filtering stage is to assist the formation of sensitive and concise representations of image texture, producing sparse feature arrays (Figure 1.5). Unlike previous haphazard procedures, we adopt principled strategies for combining measurements from many filter channels, based on response moments and parameterised Gabor signatures, and propose two efficient feature-extraction algorithms (Chapter 4). Both adapt easily to adjustments of filter characteristics, and offer the potential for elementary abstraction of imaging parameters.

Real textures are inherently variable, and their stochastic behaviour is described conveniently by random field models. The influence of spatial context is acknowledged explicitly by Markov models, which are specified by the probability density of each pixel conditional on its local image neighbourhood (Chapter 5). We review the theoretical basis of this approach, discuss manipulation of Markov models and their application to image processing, and conclude that auto-normal models have particularly attractive properties. Our proposals for a hybrid Gabor-Markov framework address several limitations of conventional approaches.

Computational considerations often dictate the use of auto-normal Markov models in practice, and we adopt this form for our experimental investigation. Orientation

is an important texture feature but deviates significantly from the normal model, and we propose angular field models to overcome this disparity (Chapter 6). In order to compare the performance of Gabor-Markov models with a conventional analysis, we adopt empirical classification accuracy as a benchmark, applied to a selection of Brodatz textures (Figure 1.6). Similar accuracy is obtained with both paradigms, but Gabor-Markov classifiers are more robust to image noise and blur, and less computationally demanding. Our experimental appraisal of Gabor-Markov models leads us to propose two families of *Sampled-Markov* models, employing concise representations derived directly from the image data. These enjoy greater robustness and higher computational efficiency than either Gabor-Markov or conventional approaches. Smooth-Sampled Markov models achieve highest accuracy in our tests, and are the preferred choice.

Classification accuracy correlates strongly with the level of correspondence between image data and the auto-normal model, and our explanations for observed performance trends are founded on this relationship (Chapter 7). This link establishes the importance of appropriate pre-processing of image data prior to statistical analysis. Assessing the significance of this thesis, we conclude that both Sampled-Markov and Gabor-Markov models meet our research objectives (page 8), and offer significant practical benefits compared to current approaches (Chapter 8). Our suggestions for further improvements may enhance the advantages of these novel representations, and we comment briefly on their potential for practical applications.


 FIGURE 1.6: Our Brodatz image set. (*Continued overleaf*)

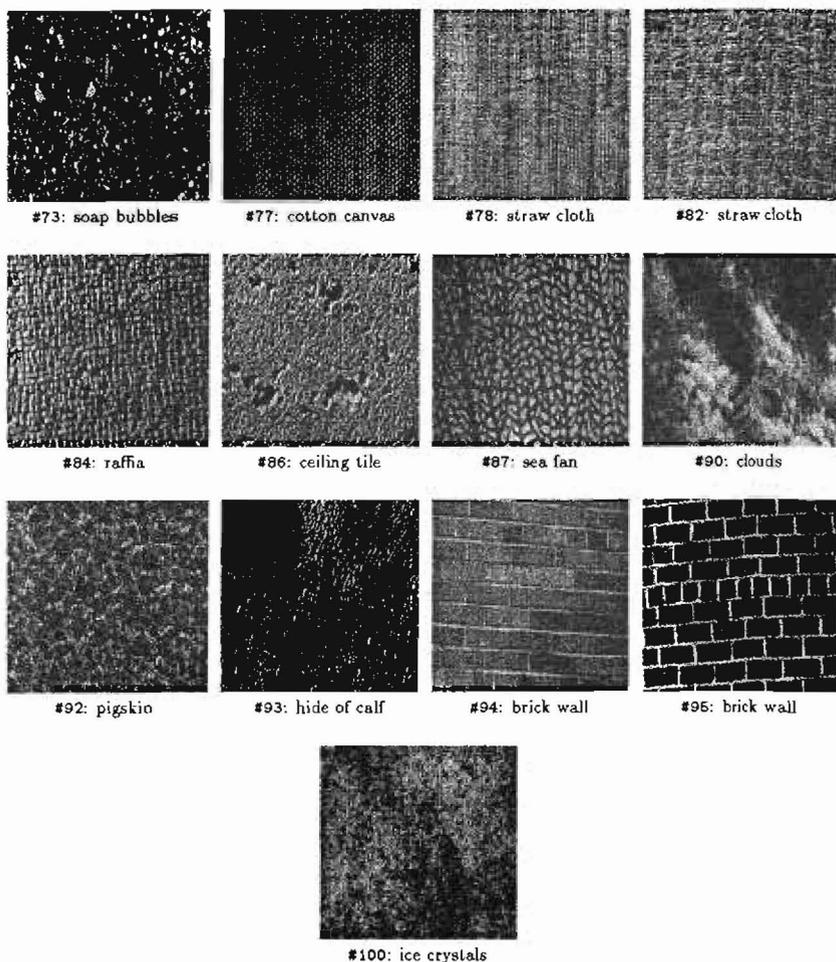


FIGURE 1.6 (*Continued*): Our Brodatz image set. Each image is 256×256 pixels and was digitised from the Brodatz (1966) album.

2

Models of Visual Attention: Serial and Parallel Paradigms

Human vision has been studied by psychologists for over a century, during which time many empirical theories have been proposed, refined, and ultimately discarded. Systematic manipulation of artificial display parameters has allowed interactions between perception and image organisation to be studied by series of psychophysical experiments. While some aspects of human visual processing are controversial, it has long been recognised that low-level vision is aptly described by an attentive-pre-attentive dichotomy employing serial and parallel processing paradigms. Current models propose that rudimentary pre-attentive analysis guides *selective* and sequential scrutiny of local image regions, marking locations corresponding to first-order differences in simple image “features”. This operation corresponds closely with texture segmentation (Chapter 3). We discuss leading psychological theories and review important aspects of the experimental data which they describe.

Unfortunately, computational analysis of loose verbal theories is not practicable, and empirical patterns of behaviour cannot directly influence the development of machine vision systems. An appropriate computational strategy and the nature of suitable image representations may, however, be suggested by informal analysis: visual attention arises as a consequence of the requirement for selective processing; and dual-paradigm processing reflects the application of abstract design principles. Proposals for many machine vision systems have been influenced by these conclusions, and first-order feature differences and dual-paradigm processing are now pervasive (Chapters 3 and 4).

2.1 Low-Level Human Visual Processing

Despite enormous progress in the last decade, computer vision still lags human vision by a large margin, particularly in respect of robustness, to image variability and degradation from noise, and speed of operation. Computer hardware is fast and reliable, and special-purpose designs may be produced to meet specific requirements. By contrast, biological components are slow and unreliable, and cannot be readily redesigned to fit a particular environment. Despite this, the human visual system not only displays superior competence but often accomplishes in real time¹ what machine vision can hardly achieve at all.

A major distinction between biological and machine vision systems is the degree of parallelism. Human brains contain very many processing elements (perhaps 10^{11}) and yet more communication channels (10^{14} ; Barlow, 1986). Simple calculations comparing reaction times with neural transmission delays confirm that extensive parallelism is essential for a significant proportion of the brain to influence responses. By contrast, many computers are parallel only to a tiny degree, and it is possible to argue that the advent of suitable massively-parallel hardware will enable the attainment of levels of performance comparable to biological vision systems, at least in the temporal domain.

Contrary to this simplistic argument, psychologists claim that while low-level perception usually appears subjectively to be immediate and complete, human vision often performs *sequential* processing, "attending" to image areas sequentially (Julesz, 1981; Treisman & Gelade, 1980). This has become known as the *attentive-pre-attentive* dichotomy (§2.1.1). Further doubt is cast on the supremacy of parallelism by theoretical arguments that it is infeasible to accomplish generic visual tasks by purely parallel processing (Gurari & Wechsler, 1982; Tsotsos, 1987). Although parallelism is certainly important, it cannot compensate for a poor understanding of the computational nature of the problem, and is not a panacea. Human vision appears to achieve instantaneously what is theoretically not computable by making implicit assumptions and approximations. These may be observed by their failures, taking the form of visual illusions or sequential processing.

In this Chapter, we review empirical evidence and psychological models of human visual processing. There is general agreement that a virtually knowledge-free spatially-parallel stage ("pre-attentive vision") guides the allocation of more sequential ("attentive") analysis.² Regions which are homogeneous with respect to a particular image property may be treated as a whole, despite possessing hierarchical structure (§2.3.4). Study of the human visual system suggests the nature of heuristic constraints that might be employed to guide selective processing, and some properties of the local image features used to form early image representations. Empirical models of human vision have often motivated approaches to machine vision (Chapter 3), and suggest reasonable objectives.

¹Generally taken to mean monitor refresh rate: 25 Hz.

²It is not helpful to define abstract terms such as "consciousness" or "awareness", and we refer to "attention" in a purely perceptual sense.

2.1.1 Evidence for an Attentive-Pre-Attentive Dichotomy

The distinction between an early "parallel" and a subsequent "sequential" stage in visual information processing has long been accepted by psychologists, and neither the existence of parallel processing at near-retinal levels nor the expression of a response as a sequence of motor actions is controversial. Although anisotropy of the retina *could* lead to serial processing stages accompanied by shifts in fixation, allocation of attention may be decoupled from eye movements, breaking this putative link (Eriksen & Murphy, 1987). Direct observation of parallel or sequential operation is clearly impossible, and these attributes must be deduced from *indirect* evidence, supported where possible by theoretical justification. Given the nature of the implementation technology, "parallel" and "sequential" can at best be approximate descriptions but are useful in reasoning about processing strategy and performance.

Visual perception has attracted prolonged research interest, and literature abounds on the subject. Differences of interpretation remain, but recently broad convergence on the major features of a model for low-level human visual processing has been reached (§2.4.1). We introduce these below (§2.2), without attempting to distinguish carefully the minor differences which exist between a number of similar theories. Refinements have been proposed over time; in order to pursue the argument, we first introduce each theory in the form in which it was originally proposed.

Behavioral evidence for an attentive dichotomy has been gathered using two main paradigms: *texture segregation* and *visual search*. In the former case, an artificial display is partitioned between two or more texture types, each formed from well-separated discrete elements. In the context of our earlier discussion (Chapter 1), these images decompose easily into micro- and macro-texture. When a difference in a "basic" image property³ occurs across the boundary, the two textures are segregated immediately and effortlessly: the boundary appears to "pop out" of the display. Conversely, when textures possess more subtle differences, they may be discriminated only by slow, careful scrutiny, even though their individual elements may be quite distinguishable in isolation. Properties mediating immediate discrimination include: colour, brightness, size, orientation, motion and stereoscopic disparity (Beck, 1983; Julesz, 1981, 1984; Treisman & Gelade, 1980). Each of the models discussed below assumes a slightly different alphabet but all generate essentially similar predictions (§2.2).

Visual search tasks have also been widely used. Subjects are presented with a multi-element display similar to those used for texture segregation (*e.g.* Figure 2.5), and are asked to decide whether a target element is present or absent. When the target possesses a basic feature not present among the other display elements ("distractors"), its presence may be determined in a time almost independent of the total number of elements ("display size"). Informally, it appears again to pop-out of the display (Treisman & Gelade, 1980). If no such description of the target is possible, perhaps because all its basic features are also held in different combinations by the distractors, then search is slow and effortful, and reaction time (RT) increases linearly with display size (Figure 2.1). Whilst it is commonly assumed that increasing RT is caused by additional processing stages, this need not necessarily be the case, and could result from

³Characteristics of these "basic properties" are discussed in §2.3, but their computational form has never been elucidated.

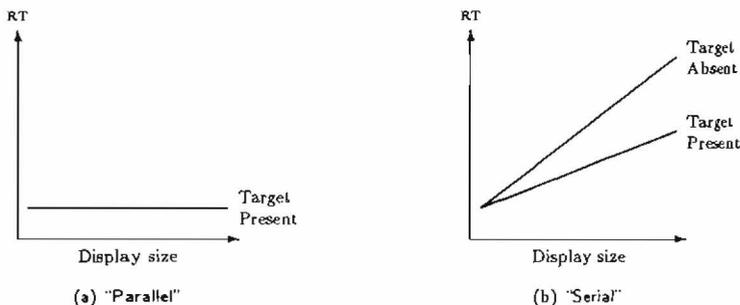


FIGURE 2.1: Schematic attentive and pre-attentive response curves. Ideal dichotomy between attentive and pre-attentive search of artificial multi element displays. Reaction time (RT) is shown as a function of the number of elements. (a) when the target is distinguished by a unique feature, it “pops out” after an interval independent of the number of distracting elements. (b) the target is discovered only by a serial scan of the display, and response time increases linearly with the number of distractors. Target-present and target-absent slopes are in the ratio 1:2.

“noise” or interactions with visual memory (Broadbent, 1987). One set of image features is sufficient to describe performance in both visual search and texture segregation tasks.

2.2 Models of Pre-Attentive Human Vision

Many theories have been advanced to explain the attentive-pre-attentive dichotomy, from behavioral, theoretical and physiological viewpoints. A common failing is their definition in ambiguous verbal terms: different interpretations may lead to conflicting predictions (Broadbent, 1987). There is often a tendency to be rather simplistic, defining a procedure suitable for processing an artificial multi-element display but which is inappropriate for the full complexity of arbitrary scenes. In particular, display elements and their properties are often assumed to be discrete whereas in practice neither is true, and it is not clear how such models could operate robustly with irregular or variable textures. Vague verbal theories may be too weak to generate testable predictions, instead forming a re-description of existing empirical results, but these shortcomings are avoided by a truly *computational* theory (Winder, 1988b, 1990).

Perceptual theories may be divided broadly into two groups. *Early-selection* models hold that comparatively little is achieved by the “parallel” stage, and that linear response functions observed for more complicated displays reflect sequential “attentive” processing of the field (Treisman & Gelade, 1980). “Perception” is achieved only for a small proportion of the scene at a time. Conversely, *late-selection* theorists argue that the parallel “pre-attentive” stage should more correctly be termed “pre-conscious”, and does in fact build a complete representation, possibly containing minor inaccuracies. The serial bottleneck occurs only during the exchange of information with conscious or motor processes — it is almost as if the visual system had been designed in isolation,

interfaces to other functions being added as an after-thought (Duacan & Humphreys, 1989). Despite their apparently opposing viewpoints, both sets of theories seek to explain the same set of experimental data, and as this has grown, the differences between them have diminished to the point where convergence has almost been reached (§2.4.1).

Pre-attentive processing is assumed to be data-driven, little affected by higher-level knowledge or expectation, and its primary purpose is to form a representation of the image at a primitive level of abstraction. Purely linear transformations of the retinal response preserve its information content but are invertible and hence do not process the data. A number of influential theories have been developed proposing similar forms for the pre-attentive representation, and hence broadly similar goals for pre-attentive processing. They are introduced below in their original form; some revisions have since occurred. Despite appearing to operate within an artificial framework, the principles guiding these models are very similar to those addressed by texture segmentation algorithms in machine vision systems (Chapter 3).

2.2.1 Texton Theory

Julesz (1980, 1981, 1984) proposed that visual input is represented in terms of an alphabet of texture primitives, which he called *textons*. These are detected at each point in the visual field by a spatially-parallel pre-attentive system which computes their local density. Differences in texton density are recorded in the pre-attentive representation, effectively forming a primitive image segmentation. Texton theory may be summarised by two heuristics:

1. Human vision operates in two distinct modes: pre-attentive and attentive. In the former, the visual field is processed instantaneously, in parallel, independent of the number of elements or complexity of the scene, over a wide visual area. By contrast, attentive processing is slow, sequential and effortful; and operates only within a restricted aperture.
2. The pre-attentive system evaluates spatial texton density and directs attentive vision to locations where differences of texton density occur. All other functions are performed by the attentive visual system.

Textons must be selected carefully: too many, and demands for attentive scrutiny arise frequently, overloading the system; too few, and pre-attentive figure-ground segregation is achieved only rarely. Proposed or accepted textons include: colour, angular orientation, width, length, binocular disparity, motion disparity, flicker rate, brightness, line terminators and line crossings. It is possible that different sets are appropriate for human and robot vision, or for different visual environments (Julesz & Bergen, 1983).

Textons of each type are processed independently by a modular pre-attentive system, which preserves only their density. Even the relative positions of textons of the same type are lost, unless some chance alignment gives rise to a new texton property (Beck *et al.*, 1983; Treisman & Paterson, 1984), confirmed by psychophysical experiments (Figure 2.2; Julesz & Bergen, 1983). The pre-attentive representation is a primitive boundary map (Sagi & Julesz, 1984, 1985), and when this alone provides sufficient evidence for a

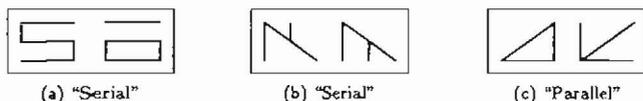


FIGURE 2.2: Search elements with similar textons. Performance at detecting one element from each pair among 35 distractors of the other type is correctly predicted by differences in texton density (Julesz & Bergen, 1983). (a,b) pairs of elements differ only in spatial arrangement, not in texton density: serial search is required. (c) these elements possess different textons: the target pops out pre-attentively.

response in an experimental judgement, or when its efficiency at directing attention to the critical location approaches 100%, “parallel” behaviour is observed. Texton theory is concerned mainly with the capabilities of the pre-attentive system: the role of attention is not clearly stated. Note that texton theory shares many characteristics with “statistical” approaches to texture analysis (§1.2), and has motivated the development of machine vision systems of this type (Chapter 3).

2.2.2 Primal Sketch

Texton theory shares many components with Marr’s “primal sketch” theory of visual information processing. Neither is a computational theory, but some aspects of the primal sketch are described more precisely. A *symbolic* retinotopic description of the scene is produced by simple filtering operations in a data-driven manner (Marr & Hildreth, 1980). Marr proposed parameterised primitives similar to those accepted as textons: edges, lines, bars and “blobs”, chosen partly for their projection-invariant qualities (Marr, 1976, 1982). A retinotopic representation called the *primal sketch* is preferred to more abstract organisations on the grounds that many of the processes which operate on it are local, due ultimately to spatial coherence of natural scenes. Texton theory is rather vague about the interaction of the pre-attentive representation with higher-level processing, whereas the primal sketch forms a complete intermediate representation from which all subsequent information is derived.

Marr proposed that pre-attentive texture discrimination occurs as a result of first-order feature differences (Marr, 1976), similar to texton theory. The raw primal sketch is a primitive abstraction of the image and cannot provide access to higher-order information in the form in which it is required. A hierarchy of representations sits above the raw sketch, each at a higher level of abstraction, transforming a large and unwieldy collection of data into a compact and relevant form. Successive abstractions are computed from more primitive representations, but do not refer back to the image data directly (Figure 2.3; Marr, 1976). Grouping adjacent image regions with compatible features is an important part of this process (§2.3.4).

2.2.3 Feature-Integration Theory

In a series of papers published over the past decade, Treisman has expounded the “feature-integration” theory of human visual attention (Treisman & Gelade, 1980; Treis-



FIGURE 2.3: **Distinction between implicit and explicit grouping.** Groups may be formed in the primal sketch implicitly by spatial-frequency filters or explicitly by symbolic grouping. (a) two blobs may be linked explicitly because of their proximity and similarity; (b) or implicitly by the action of a large-scale bar detector. Marr (1976) proposes the former.

man & Gormican, 1988). Most of the experimental support for this model has been derived from artificial visual search tasks. The definition of feature-integration theory is conched largely in the language of psychology, forming a loose framework rather than a testable theory. Key components are the automatic extraction and independent processing of elementary features (Figure 2.4), and feature-integration theory may be summarised by two assertions:

1. The retinal image is first processed in parallel along separable feature dimensions by independent perceptual analysers. Among the image properties extracted at this stage are: orientation, colour, brightness, direction of motion, and spatial contrast; and each attribute is represented in an independent feature space. This first stage is automatic, operates over a wide visual area, and has been identified as “pre-attentive”.
2. In order to perceive physical objects as unitary forms, primary features must be synthesised into *conjunctions*.⁴ Attention is required for correct identification and conjunction of an object’s features. This is a serial process which operates over a restricted visual area.

A further task of the pre-attentive stage is to mark feature boundaries in a retinotopic “map of locations” (Figure 2.4), corresponding to the primitive boundary maps envisaged elsewhere (§2.2.1). The novelty of feature-integration theory lies in its explicit identification of a role for attention: it adds detail to the regions marked by the pre-attentive stage, and detects higher-order properties. This is essential because boundaries marked in the Map of Locations are not tagged with the identity of the feature whose gradient they mark. A further facet of the model is the “Pooled Response”, which permits a global measurement of the level of activity within each feature space to be obtained quickly without the need to attend each display location sequentially.

2.2.4 Experimental Verification of Proposed Models

Each of the models described above (§§2.2.1–2.2.3) attempts to explain the same body of experimental data, and hence it is not surprising that similar predictions are made. Unfortunately, the verbal exposition of the theories means that their predictions are

⁴For example, colour and orientation are separate features, and hence a blue vertical line is represented by a conjunction of “blue” and “vertical”.

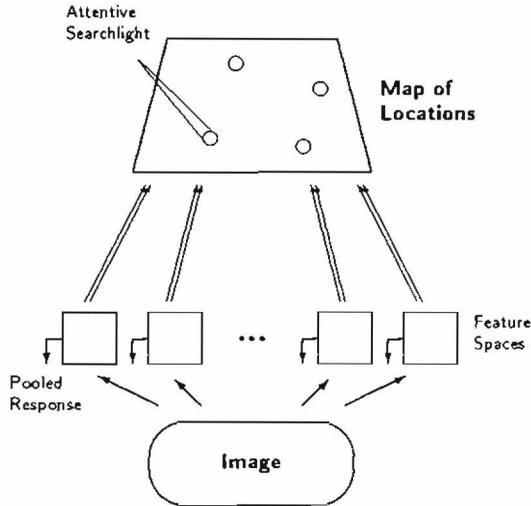


FIGURE 2.4: Schematic diagram of feature-integration theory. The image is decomposed independently in parallel along primitive feature dimensions. These mark feature discontinuities in the Map of Locations, forming a primitive boundary map. Attention is guided to these sites. A “pooled response” output is also formed, measuring the average level of activation of each feature. Adapted from Treisman (1985).

qualitative rather than quantitative, but we will attempt to circumvent this failure by restricting our comments to the most definite trends. Feature-integration theory is a little more explicit about the purpose of attention, and hence makes more detailed predictions, but we shall treat all three theories as a whole.

Visual Search

Visual search is a popular paradigm with psychologists: trials are straightforward to conduct, displays may be constructed easily, and a quantitative performance measure is obtained. Each model predicts that a target defined by a unique feature pops-out of the display (search time virtually independent of the number of distractors) in accord with the central dichotomy (§2.1.1), consistent with observed behaviour (Treisman & Gelade, 1980; Treisman & Gormican, 1988). For example, a single blue letter was found among brown and green letters with a response slope⁵ of only 3 ms per distractor, and similar responses are observed when a number of targets are present. Displays must be present for at least a minimum “initialisation” interval before anything may be perceived reliably, but a display duration of only 30–45 ms is sufficient to detect a vertical line among 35 horizontal lines (Bergen & Julesz, 1983). Perceptual pop-out is robust: reaction

⁵Reaction times often follow the pattern illustrated in Figure 2.1, and are characterised by a “response slope”, commonly measured in ms per distractor.

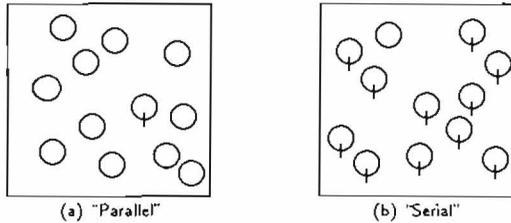


FIGURE 2.5: Artificial target-present displays used to demonstrate search asymmetry. A feature is most effective at causing the target to “pop-out” when it is present in the target but not in the distractors (Treisman & Souther, 1985). (a) the target possesses a unique feature and pops out; (b) the target is defined only by the absence of the line segment, and is detected after a serial scan.

time is rarely completely independent of display size but the slope is typically a few ms per item, an order of magnitude less than “serial” responses. A feature difference is most effective when present in the target but absent from the distractors (Treisman & Gelade, 1980; Treisman & Souther, 1985), illustrated by Figure 2.5.

A very different trend emerges for conjunctively-defined targets, where no first-order difference discriminates target from distractors. Finding T_{green} among equal numbers of T_{brown} and X_{green} required 28 ms per distractor (Treisman & Gelade, 1980). Search for an L among Ts is serial, whereas a slanted T pops out (Figure 2.6). All aspects of spatial relations demand serial processing (Treisman & Gormican, 1988) including the binding of textons to particular objects (Julesz, 1984; Treisman & Gelade, 1980). “Serial” processing is assumed when the response slope exceeds 10 ms per item. Pooling data from many sources, Treisman and Gormican (1988) found that 99% of the response variation was linear, with a target present-absent search ratio of 0.53: strong evidence for the operation of a serial self-terminating search strategy.⁶

Texture Discrimination

Texture discrimination tasks require subjects to detect a *region* rather than a single item, for example they may be forced to choose whether a boundary runs horizontally or vertically. As noted above (§2.1.1), the boundary is immediately apparent when adjacent regions have differing feature densities (Beck *et al.*, 1983; Julesz, 1984; Treisman, 1985). By contrast, if they do not have disjoint features, or have the same texton density, the boundary may only be traced out one element at a time (Treisman & Gelade, 1980), even if individual texture elements may be discriminated easily in isolation (Julesz, 1984). When a texture difference arises purely because of spatial relations, such as R and \mathfrak{R} , individual scrutiny is always required (Figure 2.2).

⁶Each display element is checked in turn, until the target is found or none remain. If search is undirected, half as many steps are required, on average, when the target is present rather than absent.

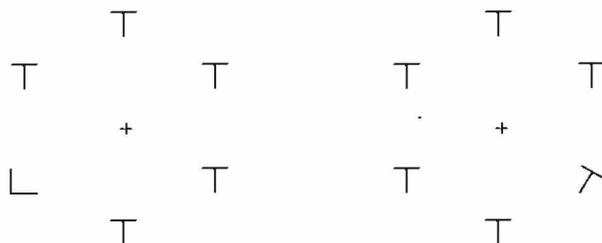


FIGURE 2.6: **Artificial search displays.** Search displays of the type used by Beck and Ambler (1972). The L in the display on the left is found only after checking each element whilst the slanted-T on the right pops-out immediately.

Experimental Procedure

Experimental procedure for both visual search and texture segregation paradigms must be designed carefully to ensure that differences in performance do indeed arise because of the influence of display conditions rather than some unintended effect. In order to eliminate eye movements from consideration, it is common to expose the display for a brief instant, typically 100 ms, during which time saccades cannot occur (Bergen & Julesz, 1983; Sagi & Julesz, 1985). Acuity variation (Anstis, 1974) may also be factored out by the use of a circular display (Beck & Ambler, 1972). A pre-cue may affect performance considerably, even if it contains no useful information for the task (Eriksen & Murphy, 1987). Display elements are usually selected purely for their perceptual properties, and letters are a poor choice because of uncertain differences in their physical appearance and associations (Gilmore, 1985). Despite these complications, the serial-parallel trends noted here have been observed by many researchers using different experimental procedures, and are beyond dispute.

2.3 Characteristics of Human Texture Primitives

Identification of the “basic image properties” of pre-attentive vision is required before a computational analysis may be conducted. Several attempts have been made to isolate definitions for Beck’s (1983) “local image properties” and Julesz’s (1981) “textons” in order to understand better the process of texture decomposition. More formal specification would also be helpful to provide guidance for the design of robot vision systems because the vagueness of verbal theories is not amenable to computational analysis (Broadbent, 1987).

One study attempted to match computational definitions to six verbal terms describing seemingly important visual properties for texture discrimination: coarseness, contrast, directionality, line-likeness, regularity, and roughness; but was only partially successful in modelling human performance (Tamura *et al.*, 1978). A more promising level of enquiry has been to abandon the search for detectors corresponding directly to high-level verbal quantities, and to seek families of simple linear filters consistent with the characteristics of simple cells. A generic three-stage region-growing algorithm

was proposed by Caelli (1985). He suggested: linear convolution followed by a point non-linearity to extract features; linear smoothing or local averaging to improve the reliability of feature estimates; and region formation by feature clustering and refinement. Coggins and Jain (1985) proposed a combination of oriented and isotropic filters, and claimed good performance with Brodatz textures. Their system was not fully automatic, and required prior knowledge of the number of texture types. Malik and Perona (1990) use linear DOOG⁷ filtering and non-linear inhibition to collect image properties, and detect boundaries by Laplacian of Gaussian filtering of feature responses. Their model successfully ranks textures with ambiguous texton statistics in the same order of discriminability as human subjects, but is not fully automatic and cannot predict some important perceptual effects (*e.g.* search asymmetry).

Recently, it has been noted that receptive field properties of simple visual cortical cells are consistent with Gabor filters (Daugman, 1985), prompting suggestions that many low-level visual effects might be explained by initial filtering with Gabor kernels (Fogel & Sagi, 1989; Sutter *et al.*, 1989). Gabor filtering models appear to overcome some current anomalies of texton theory (Nothdurft, 1990), and the attractive properties of Gabor kernels have also aroused interest from the machine-vision community. We shall postpone discussion of our experiments with Gabor filters until Chapter 4.

2.3.1 Image Features are Local

It is important to note that each of the above models proposes that pre-attentive vision is a *local* process (§§2.2.1–2.2.3). Texture primitives are compact (*e.g.* “line crossing”) and only local feature differences are computed, rendering global processing unnecessary. Direct evidence for short-range computation of feature differences comes from a textured Craik-O’Brien-Cornsweet illusion (Muller, 1986): ease of perception of a boundary between regions of different line orientation depends primarily on the difference in slope at the boundary rather than the difference between the regions as a whole. Hybrid region-edge algorithms may also be used (Mumford *et al.*, 1987). Textons capture only local spatial arrangement, and graded differences in texton properties are detected more efficiently when they are nearby (Sagi & Julesz, 1984). Anisotropic stretching alters local texton density (Figure 2.7), and discrimination performance is affected accordingly (Enns, 1986; Julesz, 1986).

2.3.2 Original Julesz Conjecture: Second-Order Statistics

In a precursor to texton theory, Julesz (1962) originally proposed that spontaneous texture segregation is mediated by differences in first or second-order statistics. At first, the theory seemed successful, and perception of texture boundaries in synthetic Markov textures was predicted correctly. Weak counter-examples were later found by synthesising textures from pairs of “iso-dipole” primitives, and Julesz (1975, page 40) acknowledged that “it would be going too far to believe that texture discrimination depends entirely on the statistical properties of textures alone”. Pratt, Faugeras and Galgalowicz (1978) developed autoregressive texture models that permitted control of

⁷Difference Of Oriented Gaussians: similar to anisotropic Laplacian of Gaussian.

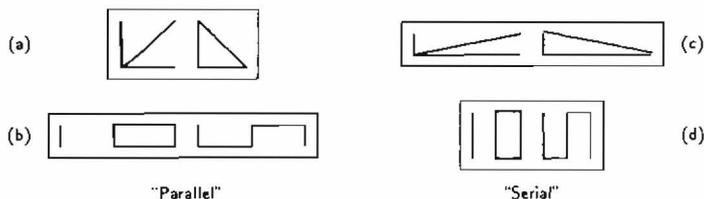


FIGURE 2.7: Effect of anisotropic stretching on texton density. Textured displays were formed by embedding a region consisting of one type of element within a background formed from the other. (a,d) original texture pairs: pair (a) segregate pre-attentively but (d) do not. (b,c) after anisotropic stretching by a factor 5:1, the position is reversed: pair (b) segregate pre-attentively but (c) do not. Affine transformation does not change the texton count for each element but *does* affect the computation of texton density (Julesz, 1986).

the statistical properties of more realistic textures than the binary dot arrays used by Julesz (§3.2.5). They noted that Julesz had omitted to define the *degree* of dis-similarity that could be perceived, and found this to be quite large for first-order properties. They concluded, however, that “the sufficiency of Julesz’s conjecture is confirmed for fields possessing spatial correlation” (Pratt *et al.*, 1978, page 801), and later confirmed this belief for “non-contrived” textures (Faugeras & Pratt, 1980).

Increasing numbers of exceptions were later discovered, however, and Julesz eventually admitted that the iso-dipole requirement was inadequate (Julesz *et al.*, 1978). He initially supplemented his model by local “bar” and “terminator” detectors (Julesz, 1980), designed to accommodate evidence that some iso-dipole textures may be discriminated effortlessly. An alternative explanation is that the “exceptions” were caused by the non-ergodicity of random fields used to synthesise test images, so they differed in *local* second-order statistics. A model based on local spatial averages of second-order statistics was proposed to reflect this possibility (Gagalowicz & Ma, 1985).

Julesz later abandoned the iso-dipole conjecture entirely in favour of texton theory, as stated in §2.2.1. Nonetheless, many experiments performed by Julesz and others show that dipole statistics *are* able to capture much important texture information, and this has motivated the development of many robot vision systems (§3.2). A further advantage of first and second-order statistics is their theoretical and computational simplicity.

2.3.3 Importance of Spatial Location

Each of the models discussed above (§2.2) proposes that image information is initially represented topographically, and it is implicitly implied that attention is mediated spatially. Several lines of evidence provide support for this possibility.

Visual Orienting

One of the simplest visual tasks is to respond to the presence of a single bright dot against a blank field. Surprisingly, detection of dot onset is influenced by a spatial pre-cue giving advance warning of its location, and an invalid cue depresses performance (Posner, 1980; Posner *et al.*, 1980). Once an image location has been "attended to", it is temporarily inhibited, and detection of a second target at the same site is slower than elsewhere (Posner & Presti, 1987). A variety of experiments have been devised to probe this effect more closely, tracking the allocation of attention across the visual field (see Winder, 1988b).

Effect of Advance Cue on Visual Search

According to the theories mentioned above (§2.2), a spatial pre-cue should have little benefit during a pop-out search because the location of the target (if present) should already be marked pre-attentively, and this prediction has been confirmed experimentally (Ambler & Finklea, 1976; Treisman, 1985). There was a small benefit on target-absent trials, perhaps because subjects are less inclined to perform a serial check before responding negatively. By contrast, conjunctive targets were detected much more easily after a spatial pre-cue, consistent with failure of the pre-attentive system to mark the target location adequately. Further experiments using similar paradigms confirmed the importance of image location to conjunctive but not single-feature targets (Nissen, 1985; Treisman & Gelade, 1980). No evidence was found that attention could be confined to a single *property* (e.g. red), rather than location.

When subjects were unable to process the display fully because of time pressure, they sometimes reported "illusory conjunctions" — perception of the correct features but in the wrong combinations (Treisman & Gelade, 1980). This motivated speculation that feature spaces might communicate only under the influence of attention, perhaps because they had parametric internal representations (Ballard, 1984), but experiments of this type are unreliable because purely perceptual effects cannot be isolated.

Detection and Identification

Further suggestive evidence for the topographic organisation of early pre-attentive representations follows from a distinction between detection and identification of a target. According to theory (§§2.2.1–2.2.3), subjects may rapidly *detect* locations of feature discontinuity, but may only *identify* the elements responsible after attending to the marked locations. This pattern of behaviour is indeed observed in practice, using displays of the type shown in Figure 2.8 (Sagi & Julesz, 1984). When the distractors are themselves non-uniform, but do not share features with the target, visual search and texture segregation both become more difficult (Callaghan *et al.*, 1986; Pashler, 1988).

The experiments reported here, together with numerous others (Winder, 1988b), suggest that image locations of feature differences are of key importance in the early parallel representation, in accord with texton and feature-integration theories. Real scenes generally have a more complex structure than artificial displays, and may not be decomposed unambiguously into independent elements, or micro- and macro-structure

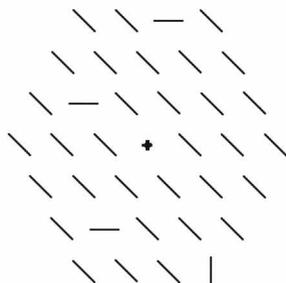


FIGURE 2.8: **Mixed-target display** used in combined counting-identification task. Subjects were required to count the number of non-distractor lines and identify their orientation (Sagi & Julesz, 1984). Detection and counting of the odd elements could be achieved in a constant period whereas identifying them accurately required a serial check.

(§1.2). In the next section, we note some modifications to the monolithic pre-attentive representation designed to accommodate hierarchical image structure.

2.3.4 Hierarchical Representations and Perceptual Grouping

Artificial displays usually consist of well-separated elements, and structure is present only at a single level (for example, Figure 2.6). In practice, real scenes may be much more complex, with ill-defined or contradictory borders (Figure 1.1 on page 2). Primitive elements at one level may be grouped together to form a meta-primitive at the next, and many levels of structure may be represented by a hierarchy. Re-examination of Figure 2.6 shows that it also possesses two levels of structure, appearing at the higher level of abstraction as two adjacent circles. Texture segregation demands a high-level description in terms of homogeneous regions rather than individual primitive elements.

When we search artificial displays having a clear hierarchical structure, as shown in Figure 2.9, we are able to break the image into distinct groups pre-attentively. Reaction times are linear in the number of *groups* rather than elements, suggestive of semi-serial search: each group may be checked in parallel using the pre-attentive mechanism, but only one group at a time (Treisman, 1982). Valid perceptual groups may only be formed from contiguous regions sharing a common feature.

Spatial Frequency Grouping

A very simple explanation for the phenomenon of hierarchical grouping is a coarse-to-fine image scan involving the use of tuned spatial-frequency channels (Wilson, 1983). An example of how this might operate is shown in Figure 2.10: the two groups emerge immediately after filtering with a Gaussian kernel. Increasingly detailed examination requires the use of filters with smaller space-constants, and the MIRAGE model proposes non-linear summation to avoid loss of resolution (Watt, 1988). It has been suggested that output of simple linear filters could underlie texture segregation (Bergen & Adelson, 1988)

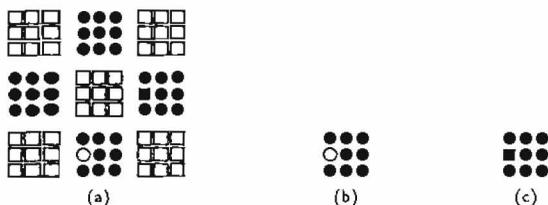


FIGURE 2.9: Artificial image with two levels of structure. (a) hierarchical decomposition of this image is straightforward. (b,c) these odd elements are often missed in (a) at first glance but are salient within each sub-group in isolation.

and Gestalt grouping (Reed & Wechsler, 1990). Perceptual groups are still effortlessly perceived in displays similar to Figure 2.10 after filtering to eliminate low spatial-frequencies (Jañez, 1984; Julesz & Kröse, 1988), and so this must be at most a partial effect. Many grouping strategies are possible, ranging from implicit grouping by linear filtering to explicit symbolic linking (Beck, 1983); further discussion appears in Winder (1990).

Attentive-Pre-Attentive Continuum

Multiple-resolution image representations blur the distinction between attentive and pre-attentive vision. An initial “parallel” stage partitions the display into a number of regions, each of which may be selected (“serial”) and further decomposed. Algorithms of this type are semi-serial, consisting of a sequence of parallel operations. It is suggested that the size and shape of these regions may vary (Eriksen & St. James, 1986; Treisman & Gelade, 1980) but that they must be spatially contiguous (Crassini, 1986). This is the refined description offered by feature-integration theory (Treisman & Gormican, 1988), which also predicts an enhanced role for the Pooled Response (Figure 2.4). A similar structure has now been proposed for texton theory, suggesting that finer discrimination is possible when attention is confined to a smaller area (Bergen & Julesz, 1983; Julesz, 1986). Definitions of “texture” and “texture element” were revised to make them scale-independent, so that texton properties could be extracted at many resolutions.

Whereas texton and feature-integration theory suggest that an initially-coarse representation is progressively refined, Marr’s (1982) primal sketch involves a recursive *synthesis*. Starting with the complete raw primal sketch, representations are formed at successively greater levels of abstraction, reflecting higher-order and ultimately viewpoint-invariant properties. This strategy is more akin to structural analysis (§1.2), but the distinction is not as great as may be first thought, because it is still sensible to scan the primal sketch in a predominantly coarse-to-fine direction. With either method, the use of multiple-resolution representations allows greater benefit to be derived from the “parallel” stage when the scene has a favourable spatial organisation.

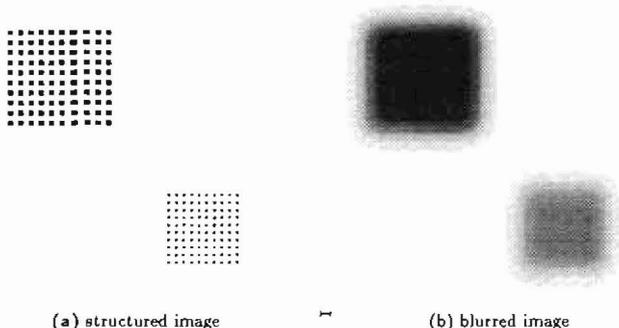


FIGURE 2.10: Formation of perceptual groups by spatial filtering. (a) an artificial image, containing two perceptual groups. (b) convolution with a Gaussian kernel (space constant as shown) has successfully picked out the two groups.

2.4 Origins of Dual Paradigm Processing

Having reviewed the main features of the attentive-pre-attentive dichotomy, and the models which seek to describe it, we now turn to causes for this behaviour, and discuss implications for machine vision. We conclude that visual attention implies a *selective* analysis of image data, and hence that division into parallel and serial paradigms reflects good design principles.

2.4.1 Guided Search Model

Recent experiments have shown that certain *combinations* of features (or conjunctions) may be detected in parallel, for example O_{red} among O_{green} and X_{red} , contrary to the dichotomy stated above (§2.1.1). Increasing numbers of special cases to the theory were proposed in the form of unified conjunction-detectors, until it was shown that *most* conjunctions were able to mediate parallel detection under suitable circumstances (Treisman & Sato, 1990; Wolfe *et al.*, 1989). Exceptions occur when the two features forming the conjunction are of the same type, such as part-red part-green (Wolfe *et al.*, 1990).

This result is an embarrassment for the bottom-up models stated in §2.2, but the circumstances under which parallel detection of conjunctions is possible are limited, and two conditions must be satisfied. Firstly, display elements must be very highly discriminable, so that “noise” is insignificant. Apparently, colours used in early tachistoscopic displays were insufficiently saturated to meet this condition (Treisman & Sato, 1990). Secondly, subjects must know in advance the exact feature composition of the target(s), and may only scan for conjunctions of a single type at a time.

A simple extension to the pre-attentive models given in §2.1.1 is sufficient to explain this result. Each feature-space marks locations of feature discontinuity in a bottom-up fashion, as before, but also compares local scene properties with those of the target

using top-down knowledge, and the sum of these signals is written into the boundary map. We might suppose that candidate locations are ranked and scanned in turn, and hence the correct target is present at the first location to be attended to when noise is insignificant. If advance knowledge is imprecise or absent, the bottom-up procedure applies. Top-down selection is not very efficient, and may easily be overcome by "noise" or imprecise specification of the target, or target-distractor similarity.

This modified framework is called Guided Search (Wolfe & Cave, 1990). A fast spatially-parallel "pre-attentive" stage forms a primitive segmentation, using both bottom up *and* top-down information, to which each feature space contributes independently. The purpose of this rather inflexible and simple-minded stage is to guide a spatially-restricted, slower, more sophisticated "attentive" processor to locations of potential importance efficiently. Effective guidance hides the serial nature of the attentive process (Hoffman, 1979). Qualitative predictions of the model have been formed by computer simulation (Cave & Wolfe, 1990), and are essentially the same as those made by late-selection models (Duncan & Humphreys, 1989).

2.4.2 Visual Physiology

Only a very brief mention of the anatomy and physiology of the human visual system lie within the scope of this thesis, but physiological approaches are important because they may motivate developments at cognitive or computational levels. Visual areas of the human (and related) brain are not amorphous but consist of largely independent functional units. Cowey (1985) mentions 12 different visual areas in monkey, and notes that new areas are discovered "... at a rate of about one every two years". Cortical maps usually follow a topographic layout, so that cells responding to adjacent image locations are nearby. Several functionally-specialised pathways run in parallel, often physically segregated from each other (Zeki & Shipp, 1988). Similar specialisation is inferred in the human brain from the selective loss of function suffered by stroke victims. The brain is, of course, very poorly understood, and it would be foolhardy to base a visual theory upon a few isolated neurophysiological observations. Nevertheless, Livingstone and Hubel (1987) had sufficient confidence in the significance of their own observations of separate processing channels to predict qualitative performance in a series of simple psychophysical experiments. Receptive field profiles of simple cells indicate what information has been extracted at the lowest level (Pollen & Ronner, 1981; Wilson, 1983), from which performance at threshold discrimination tasks has been predicted. Different representations may be used at higher levels, and wider conclusions should be approached with caution (Marr, 1982).

Single-cell recordings permit observations of a working brain, although it is not operating under normal conditions. Neural responses are found to be greatly attenuated when attention is directed elsewhere in the visual field (Moran & Desimone, 1985). Increasing the amount of attention directed at a stimulus can enhance both the responsiveness and selectivity of neurons that process it (Spitzer *et al.*, 1988). These results from monkey confirm the spatially-selective nature of attention. Actions of single neurons may be tenuously related to the computational task, however, and wider conclusions are speculative. For similar reasons, artificial neural network models of visual attention are often inappropriate (Winder, 1989).

2.4.3 Justification for Serial and Parallel Paradigms

Empirical investigation by psychologists has established some major features of the human visual system: a partial dichotomy between parallel and serial modes; local image “features”; functional specialisation; and selection by image location. Competitive evolutionary pressure has probably resulted in human vision adopting a form that is at least locally optimal, assuming Nature is not unduly capricious, and hence these observations may be relevant to the design of machine vision systems. Simply designing systems with similar external characteristics is not a helpful way to proceed because this merely redescribes the psychological data; of greater relevance is the computational logic which lies behind these external manifestations.

These questions were addressed by Marr in the construction of his primal sketch model (Marr, 1976, 1982). Marr's background was in physiology, and the operations he proposes are motivated, at least in part, by empirical studies of the visual cortex. He realised, however, that it was necessary to exercise “extreme caution in making inferences from neurophysiological findings about the algorithms and representations being used . . .” (Marr, 1982, page 26). Of central importance is the *information* that needs to be represented and the *process* that needs to be implemented. Perceptual information processing may be addressed at three levels (Marr, 1982, page 25):

1. **Computational theory:** What is the goal of the computation, why is it appropriate, and what is the logic of the strategy by which it can be carried out?
2. **Representation and algorithm:** How can this computational theory be implemented? In particular, what is the representation for the input and output, and what is the algorithm for the transformation?
3. **Hardware implementation:** How can the representation and algorithm be realised physically?

Understanding the computational level is of central importance, as this defines the ends to which levels 2 and 3 provide the means. Human and machine vision systems must operate in similar visual environments, and employ analogous physical optics, and hence may address comparable information-processing requirements.

Assuming reasonable spatial and gray-level resolution (say $1000 \times 1000 \times 256$), the number of possible visual stimuli is extremely large ($> 10^{2,000,000}$), even when the influence of context or other factors are ignored, and precludes a brute-force approach to the generation of suitable responses. It has been suggested that the pervasive perceptual tasks of visual search and image segmentation are computationally intractable (Gurari & Wechsler, 1982; Tsotsos, 1987). This limit cannot be circumvented by clever implementation techniques or parallel hardware, and applies equally to biological and machine vision systems. It implies that visual information must be processed *selectively*.

Selection of Appropriate Computational Paradigm

Several design principles have been proposed to guide appropriate choices for operations to be conducted by a sequential analysis. The maximum power–minimum cost principle

seeks to maximise an abstract performance metric, and those of least commitment and graceful degradation address the variable nature of the physical world (Marr, 1982; Tsotsos, 1987). Operations performed in a spatially-parallel fashion incur a greater hardware cost because suitable processors must be replicated for each sub-image location, while sequential processing receives a temporal penalty. An ideal system with unlimited capacity would compute in an entirely parallel fashion, but it is inefficient in practice to devote limited resources to providing parallel processors to support operations which are rarely invoked. For example, it is entirely reasonable to recognise one's grand-mother, but perhaps more cost-effective to selectively apply a single detector responding to the abstract property of "grand-motherness", rather than have a bank of processors dedicated to this infrequent operation. Generic "interest operators" are strong candidates for parallel processing, since it is essential for selective image analysis to be targeted efficiently. Since physical objects are of ultimate interest, it is sensible to select "interest operators" corresponding to putative physical boundaries. Assuming the world to be composed mainly of smooth surfaces and collections of similar objects, detection of local feature differences and grouping image parts are key operations, whereas computation of spatial or abstract relations may be deferred until a selective sequential analysis (Ullman, 1984).

To some extent, the choice of which perceptual tasks are conducted "in parallel" depends on how efficiently they may be implemented. Simple local computations which are hardly influenced by local data may be performed very efficiently by data-parallel processors, whereas data-dependent or irregular computation is better supported by the more costly task-parallel paradigm (§A.1). Both variants are required for image analysis, prompting the development of a *dual-paradigm* parallel processor, the DisPuter (§A.1.3). The high efficiency of local data-parallel processing probably lies behind the apparent importance of image location in pre-attentive human vision and the restricted influence of top-down knowledge. Domain-independent "features" may be deduced from unusual image events or "suspicious coincidences" (Barlow, 1986; Lowe, 1987), and multiple-resolution or hierarchical representations further maximise performance (Terzopoulos, 1986). Note that no advantage of a dual-paradigm strategy accrues if the "pre-attentive" representation is interpreted completely by sequential analysis — this would merely trade exponential space for exponential time. Following the principle of modular design (Marr, 1976), little has been said about processes operating during selective sequential image interpretation because these are very task specific.

2.4.4 Conclusions: Local Feature Differences Guide Selective Processing

Extensive behavioral evidence shows that low-level human vision displays a dichotomy between "attentive" and "pre-attentive" processing. Pre-attentive vision operates over a wide image area whereas attention is selective and spatially localised, although this distinction is less sharp for hierarchical displays. No benefit may be derived from simply replicating these characteristics in machine vision systems, but aspects of the computational strategy employed may be inferred by informal analysis. Current models propose that the role of pre-attentive vision is to guide *selective* attention to important or "salient" image locations by rudimentary processing of local image properties (Wolfe

& Cave, 1990). Attempts to find computational definitions for these “features” have failed, but their precise form is probably less important than their general nature. Our subjective experience of immediate and effortless perception reflects the efficiency of this dual-paradigm approach.

Computational analysis suggests that it is infeasible to process image data completely, and hence that selective interpretation is imperative (Gurari & Wechsler, 1982; Tsotsos, 1987). Vision is ultimately concerned with perceiving physical objects, and heuristic assumptions about their nature confirm that first-order differences in local image statistics form an appropriate pre-attentive representation. Abstract design principles dictate that available resources should be divided between elementary “parallel” guidance and detailed “sequential” analysis, the latter paradigm being more suitable for unusual or irregular computation. Proposals for machine vision systems are often influenced by similar reasoning (Chapter 3).

3

Approaches to the Representation and Analysis of Visual Texture

Visual texture is concerned with spatial composition and appearance, and is an inherent characteristic of physical surfaces. Texture analysis has several goals, but we shall restrict our main discussion to the location of texture edges (segmentation), and characterisation of image regions (classification). Classification is often the more straightforward task, having more constrained results, but requires texture samples to be homogeneous and to have been previously segmented from the remainder of the image.

Representations of visual texture are commonly divided into structural and statistical models (§1.2), analogous with the attentive-pre-attentive dichotomy in low-level human vision (Chapter 2). This correspondence is more than superficial, and models of pre-attentive vision have often influenced approaches to texture analysis (and *vice versa*). Statistical models share many characteristics with pre-attentive vision: they represent texture indirectly by simple local statistics; and correspond naturally with the data-parallel paradigm (§A.1). Texture features are often derived from second-order image statistics, and first-order feature differences used to locate hypothetical texture boundaries (§3.2). Conversely, structural models frequently require irregular data-dependent processing, but seek to describe hierarchical structure more completely (§3.3).

Characteristics of both paradigms may be adjusted to suit the application: segmentation requires features to represent local image properties (§3.5), whereas it is often

more appropriate during classification to describe the image as a whole (§3.4). Gabor filters achieve an optimal compromise between these conflicting requirements (Chapter 4). Texture variability and the probabilistic influence of spatial context are represented explicitly by Gibbs–Markov random field models (Chapter 5). Building on psychological models of low-level human vision (Chapter 2) and the discussion of current approaches to texture presented in this Chapter, we propose a hybrid *Gabor-Markov* framework for texture analysis (Chapter 5), combining the best aspects of statistical and structural models. In this context, Gabor filtering may be described as “pre-attentive”, and Markovian analysis as “attentive” processing.

3.1 Introduction: Texture Representations

Texture analysis is an important component of low-level vision, providing information about the nature and three-dimensional shape of physical surfaces. Image analysis is ultimately concerned with properties of the *scene*, and visual texture has many advantages in this regard: it is a fundamental property of physical objects; is transformed predictably by a change of viewpoint; and is robust to changes in lighting or other external influences. Despite its practical importance, an acceptable formal definition of “texture” does not exist (Ahuja & Rosenfeld, 1981; Haralick, 1979). The *Oxford Handy Dictionary*¹ offers “arrangement of threads in textile fabric, quality of a surface or substance when looked at” (page 942), giving two complementary alternatives. When the former is generalised, so that texture is represented by its primitive elements and their spatial arrangement, these definitions reflect the division into *structural* and *statistical* approaches noted above (§1.2). Descriptions of visual texture are commonly classified according to this dichotomy, which successfully characterises the major aspects of texture analysis, but the distinction is becoming increasingly blurred as more complex models are developed (Haralick, 1979).

“Structural” approaches to texture are concerned with its hierarchical nature, and make explicit the distinction between *microtexture* and *macrotexture*. Microtexture refers to the fine-scale residual variation present in each texture primitive, and macrotexture to their spatial arrangement. The hierarchy may be expanded recursively, with macrotexture at one level forming microtexture at the next, and very complex spatial arrangements may be represented, possessing structured organisation similar to that observed in many natural textures (Ahuja & Rosenfeld, 1981; Marr, 1976). Structural descriptions are most appropriate for regular textures, such as reptile skin (#22) and handwoven rattan (#65; Figure 1.6 on page 11), but are awkward for representing more amorphous structure, such as beach sand (#28) or clouds (#90). A further disadvantage is that analysis of many different texture primitives is computationally demanding, as illustrated above (Figure 1.2 on page 4). We justified the existence of an apparent dichotomy in low-level human vision by the requirement for selective image analysis (§2.4), and similar reasoning applies to machine vision, suggesting that it is preferable to process a single texture rather than many individual elements (Marr, 1982). “Statistical” descriptions of texture adopt this more economical approach, representing

¹Sixth Edition, Chancellor Press, London.

patterns of spatial variation indirectly by derived properties or “features”. In this respect, the partition into “structural” and “statistical” approaches is analogous to the human attentive–pre-attentive dichotomy (Chapter 2), and is often characterised aptly by “serial” and “parallel” processing.

Both structural and statistical approaches represent visual texture concisely by higher-order properties or “features”, the most appropriate form for which is influenced by the application. We shall be concerned mainly with classification and segmentation, commenting in passing on image synthesis and restoration techniques. We describe these important operations briefly in the remainder of this section in order to motivate the selection of suitable feature properties during our review of statistical and structural approaches (§§3.2–3.3). Texture classification and image segmentation algorithms are discussed in more detail below (§§3.4–3.5).

3.1.1 Appropriate Features for Texture Classification

Classification of a textured image involves assigning it to a particular class, each of which is represented by a small number of “features” derived from appropriate training data. When each class contains only a single object, this process corresponds to recognition. Images are usually assumed to be homogeneous, containing texture of a single type, and a single set of features is used to represent the entire image, enforcing global² comparison between the unidentified “test” image and known “training sets” to establish the correct class label. Abstraction away from raw pixel intensity allows accurate classification to be achieved without precise duplication of viewpoint or imaging conditions (§3.1.3). It is convenient to collect texture parameters into a feature vector, and to view the training phase as forming a partition of multi-dimensional feature space (Dejviver & Kittler, 1982). Whilst the use of global features may simplify classifier design, it implies that a hypothetical object may be classified only after a separate process has segmented it from its image surroundings. Feature extraction and classification are logically distinct, and we shall treat them separately, discussing classification algorithms in §3.4.

3.1.2 Appropriate Features for Image Segmentation

Segmentation of an image into homogeneous textured regions suitable for classification necessarily involves the use of *local* features. When an image varies unpredictably, feature estimates can be improved by extending the area over which they are collected, but this reduces the precision with which a texture boundary may be localised. This trade-off is a manifestation of the uncertainty principle of information (Wilson & Granlund, 1984), and highlights a potential conflict when the same features are used for both classification and segmentation.

Partitioning an image into regions which are homogeneous with respect to some property is usually insufficient by itself to form a good segmentation. A narrow interpretation of this description could involve clustering pixels sharing a common gray

²In this context, “global” implies a lack of spatial localisation within each image.

level, even if they are distributed in small clumps across the image: fragmented boundaries would result, with no apparent overall structure (Rosenfeld & Kak, 1982). A good segmentation makes explicit the arrangement of physical objects which gave rise to the image, rather than re-expressing image-related information. This concept lay behind Barrow and Tenenbaum's intrinsic images, which seek to represent object-related properties such as surface depth and orientation (Barrow & Tenenbaum, 1981). Ideally, this additional information would be available to assist the segmentation process, but in practice segmentation may contribute to the *formation* of intrinsic images, and no prior information may be assumed. As with pre-attentive human vision, heuristic assumptions about the scene suggest that first-order differences in simple local image features are appropriate indicators for hypothetical object boundaries (§2.4).

Segmentation is a difficult problem because the number of possible scene partitions grows exponentially with image dimensions (Gurari & Wechsler, 1982). Texture variation within each region confounds the characteristic image defining the desired partition, and this effect has been likened to the modulation of a carrier signal (Clark *et al.*, 1987). In human vision, this complexity is apparently overcome by an attentive-pre-attentive dichotomy involving scrutiny of *selected* image regions. Automatic segmentation requires accurate borders to be found in the absence of additional prior knowledge about the scene: segmentation and texture description must proceed simultaneously. Approximate boundaries may sometimes be located fairly quickly, but refinement involves intense local scrutiny and is very slow (Banz & Reinhardt, 1981). As described in §3.5, edge detection and region growing are the major approaches to this problem (Ballard & Brown, 1982), although many practical systems require additional information in the form of texture or scene parameters provided by human intervention.

3.1.3 Invariant Texture Features

Texture variation is confounded by viewpoint, lighting, surface shape, and other extrinsic factors. Ideally, these would be eliminated prior to texture analysis, but this is seldom possible and it may instead be necessary to *estimate* these parameters. Features may be designed to be invariant to certain transforms in order that reliable classification and segmentation may be obtained. For example, circularly-symmetric features are clearly rotation invariant, although they fail to capture the full structure of oriented textures (Kashyap & Khotanzad, 1986; Vickers & Modestino, 1982), and fractal features are somewhat scale invariant (Peleg *et al.*, 1984). An alternative is to employ parameterised texture models which allow viewpoint parameters to be estimated from the image (Cohen *et al.*, 1991).

Variation in lighting or surface geometry may give rise to smooth changes in first-order statistics which do not reflect intrinsic properties of the texture, and hence differences in first-order properties cannot be relied upon to discriminate textures. When these effects are global to the test image, they may be eliminated by histogram equalisation (Hsiao & Sawchuk, 1989; Pietikäinen *et al.*, 1983; Weszka *et al.*, 1976) or normalisation (Faugeras & Pratt, 1980; Tomita *et al.*, 1982); and local variations produced by shadowing may be reduced by logarithmic pre-processing (Mitchell *et al.*, 1977; Voorhæs & Poggio, 1987).

3.2 Statistical Approaches to Texture Analysis

Statistical approaches to texture analysis follow the second form of the definition on page 34, and seek to represent texture indirectly by properties or features derived from its spatial distribution of gray levels. No attempt is made to understand the hierarchical structure of the texture; only properties derived directly from pixel values are used (Ahuja & Rosenfeld, 1981). The justification for this approach is the observed variability of natural texture, such as beach sand (#28) or European marble (#63), which often makes any division into microtexture and macrotexture somewhat arbitrary (Tamura *et al.*, 1978).

Suitable image features may often be extracted very efficiently by simple local processing, and hence may be computed by data-parallel architectures (§A.1.3). Statistical approaches are analogous to the role proposed for pre-attentive human vision: the common objective is to compute an elementary representation in which first-order differences reflect object boundaries or “texture edges” present in the scene (§2.4). Models of pre-attentive human vision have influenced the design of these systems, and have suggested suitable characteristics for the alphabet of primitive texture features. It is essential for these to reflect scene boundaries accurately, because it is not possible to compensate subsequently for a loss of information at this stage. Both the power spectrum and co-occurrence methods employ second-order image statistics gathered from a wide area to increase reliability (§3.2.2; §3.2.3), and are often applied to texture classification. Image filtering generates local features more suitable for segmentation algorithms (§3.2.4), and the Gabor energy measures discussed in Chapter 4 are of this type. Texture variability is acknowledged by random field models (§3.2.5), including Gibbs–Markov models discussed in Chapter 5.

3.2.1 First-Order Image Statistics

Pixel intensity is the simplest image property that can be employed as a texture feature, and provides sufficient information to segment simple structured scenes such as single cells (Blanz & Reinhardt, 1981). Intensity distributions of adjacent textured regions may have considerable overlap, and first-order properties are not reliable texture features because they are defined without reference to image context, and are strongly influenced by the image acquisition system (§3.1.3). Other first-order properties are similarly confounded, and it is therefore necessary to seek features sensitive to the spatial arrangement of the texture, modifying their response to each pixel according to its context.

Optimal edge detectors have been developed to respond to local variation in pixel intensity, marking boundaries accurately even in noisy images (Canny, 1983; Spacek, 1986). Local edges reflect higher-order image properties, and are relatively unaffected by smooth changes in surface orientation, and hence are a more robust texture characteristic. Textures are spatially varying, and conventional edge-finders respond to many internal microedges in addition to the desired macroedges when adjacent textures do not differ significantly in mean luminance (Figure 1.2 on page 4). An attempt to classify each edge segment by assuming a particular texture model was only partially successful:

some microedges were filtered out but many spurious edge segments remained (Eom & Kashyap, 1990).

3.2.2 Global Image Statistics

First-order properties represent image location precisely but are insensitive to spatial context; at the other extreme, power spectra capture aspects of spatial variation but are insensitive to image location. Power spectra are too bulky to be used directly as texture features, and such an arrangement would also be insufficiently abstract, but may be employed indirectly by deriving features from them to represent heuristic perceptual properties such as “coarseness” and “directionality” (Ballard & Brown, 1982). Another method uses gray-level run-length statistics at various orientations, which are again collected into matrices and represented indirectly by heuristic features (Connors & Harlow, 1980; Haralick, 1979).

Both these methods are able to discriminate some varieties of natural texture, but are less reliable than other features reported below (Connors & Harlow, 1980; Weszka *et al.*, 1976). They are still defined at an insufficiently abstract level, and are sensitive to noise and monotonic changes in gray level (Haralick, 1979). Whilst their inherently global nature is suitable for image classification, it is less appropriate for accurate boundary localisation.

3.2.3 Second-Order Co-Occurrence Matrices

Julesz’s (1962, 1975) conjecture that pre-attentive human vision is sensitive only to first and second-order statistics has been extremely influential in motivating the search for reliable generic texture features (§2.3.2). Although the human visual system certainly does not perform perfectly, it is attractive as a baseline, and controlled experiments show that differences in second-order image statistics correlate closely with pre-attentive texture discriminability. Second-order statistics are defined for pairs of pixels, and hence are the simplest properties affected by spatial context.

Second-order statistics may be operated on conveniently by collecting them into a *co-occurrence matrix* (Haralick *et al.*, 1973). For a given offset vector δ , matrix elements represent the number of times each pair of gray levels occurs in the image at separation δ , and form an estimate of the second-order transition probabilities on normalisation. It is usual to restrict δ to cardinal directions and integer pixel steps, and it may be represented as (d, θ) . The rank of the co-occurrence matrix $\Phi(d, \theta)$ matches the number of allowable gray levels, K . Symmetry is implied:

$$\Phi(d, \theta) = \Phi(d, \theta + \pi)^T.$$

An intermediate set of matrices is defined along principal directions.

$$S(d, \theta) = \frac{1}{2} [\Phi(d, \theta) + \Phi(d, \theta + \pi)], \quad \theta = 0, \pi/2, \pi, 3\pi/2 \quad (3.1)$$

Circular symmetry may be enforced by summing over all orientations.

$$S(d) = \sum_{\theta} S(d, \theta) \quad (3.2)$$

Co-occurrence matrices are too unwieldy to use directly, particularly for large K , and may exceed the dimensions of the image! They are often condensed into more compact features, achieving graceful degradation when the matrix is sparsely populated. Fourteen features have been selected heuristically to express significant variation (Haralick *et al.*, 1973), of which the five most commonly used are (Ballard & Brown, 1982; Conners & Harlow, 1980; Haralick, 1979):

$$\begin{aligned}
 \text{Energy:} \quad E(d, \theta) &= \sum_{i,j=0}^{K-1} [S_{i,j}(d, \theta)]^2 \\
 \text{Entropy:} \quad H(d, \theta) &= - \sum_{i,j=0}^{K-1} S_{i,j}(d, \theta) \log S_{i,j}(d, \theta) \\
 \text{Correlation:} \quad C(d, \theta) &= \frac{1}{s_x s_y} \sum_{i,j=0}^{K-1} (i - \bar{x})(j - \bar{y}) S_{i,j}(d, \theta) \\
 \text{Homogeneity:} \quad L(d, \theta) &= \sum_{i,j=0}^{K-1} \frac{S_{i,j}(d, \theta)}{1 + (i - j)^2} \\
 \text{Inertia:} \quad I(d, \theta) &= \sum_{i,j=0}^{K-1} (i - j)^2 S_{i,j}(d, \theta) \tag{3.3}
 \end{aligned}$$

where

$$\begin{aligned}
 \bar{x} &= \sum_{i=0}^{K-1} i \sum_{j=0}^{K-1} S_{i,j}(d, \theta) \\
 s_x^2 &= \sum_{i=0}^{K-1} (i - \bar{x})^2 \sum_{j=0}^{K-1} S_{i,j}(d, \theta)
 \end{aligned}$$

and similarly for \bar{y} and s_y^2 . It is not practical to compute these features for many offset lengths, and often a single value is selected (*e.g.* $d = 1$). Haralick (1979) has suggested that there tend to be natural constraints between co-occurrence probabilities at one spatial distance with those at another, but ability to discriminate some textures depends critically on the choice of d (Modestino *et al.*, 1981). At a large separation, individual pixels tend to be weakly correlated, and the resulting statistics are noisy (Weszka *et al.*, 1976). If gray levels are compared between local *neighbourhoods* rather than individual pixels, to form modified co-occurrence matrices, this effect is eliminated. Unfortunately, the arbitrary feature definitions (3.3) fail to capture all significant texture structure (Conners & Harlow, 1980; Modestino *et al.*, 1981), and have no obvious perceptual correlates (Tamura *et al.*, 1978).

Despite these reservations, empirical and theoretical comparative studies have shown that co-occurrence methods are successful at discriminating natural textures (Conners & Harlow, 1980; Weszka *et al.*, 1976). Classification accuracy was significantly higher than power spectrum or run-length methods with Brodatz textures, aerial imagery, and synthetic images, consistent with the Julesz conjecture that second-order statistics capture most important texture variation (§2.3.2). Local co-occurrence statistics may be collected over small or irregular regions, and benchmark studies have shown that

reasonable localisation accuracy is achieved during image segmentation (du Buf *et al.*, 1990).

Concise Representations: Co-Occurrence Histograms

Co-occurrence methods suffer from two main disadvantages: a large volume of intermediate data must be processed; and the heuristic features (3.3) are arbitrary and lossy. Re-quantising the image to a smaller number of gray levels reduces the rank of intermediate matrices, decreasing computational requirements, and has negligible impact on accuracy provided a reasonable number of gray levels remain ($K = 64$; du Buf *et al.*, 1990; Vickers & Modestino, 1982). Greater efficiency improvements are obtained by compressing the $K \times K$ co-occurrence matrix into a single K -element histogram. Weszka *et al.* (1976) suggested the use of absolute gray-level difference to index this histogram, and proposed definitions for features analogous to those used with full co-occurrence matrices. Empirical classification accuracy with their histogram features was about equal to that achieved with features defined from the full co-occurrence matrix, but achieved a considerable computational saving. Conners and Harlow (1980) suggested this apparent equality might reflect the inability of common co-occurrence features (3.3) to capture all the information present in the matrix rather than a deeper equivalence.

Unser (1986) noted that the covariance matrix \mathbf{C}_{xy} of a pair of random variables, x and y , may be diagonalised by taking their sum and difference:

$$\begin{aligned} \mathbf{C}_{xy} &= \sigma_{xy}^2 \begin{pmatrix} 1 & \rho \\ \rho & 1 \end{pmatrix} \\ \begin{pmatrix} u \\ v \end{pmatrix} &= \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} \\ \mathbf{C}_{uv} &= \sigma_{xy}^2 \begin{pmatrix} 1 + \rho & 0 \\ 0 & 1 - \rho \end{pmatrix}. \end{aligned}$$

After an approximation, this justifies the use of sum and difference histograms to represent second-order statistics in place of bulky co-occurrence matrices. Unser proposed another set of heuristic features for use with these histograms; in nine out of fourteen cases they were identical to those proposed earlier by Haralick *et al.* (1973). Empirical classification results suggested that Unser's histogram features were almost as accurate as those obtained from the full co-occurrence matrices, confirming the inadequacy of the latter.

Improved Co-Occurrence Features: Stochastic Models

Histograms and matrices of second-order image statistics are both dependent on the use of heuristic features to express their information more succinctly, leading to arbitrary approximations. An alternative approach is to apply a stochastic model to all the entries, making explicit the variability of natural textures, and parameters may be selected and weighted appropriately by maximum-likelihood methods. Vickers and Modestino (1982) assume the co-occurrence matrix is drawn from a multinomial distribution, with independent entries, but note that accurate parameter estimation requires

a large amount of training data. If this is not available, they suggest additional constraints could be provided by assuming a random field model for the texture (§3.2.5). Good empirical results were reported on Brodatz classification, but a thorough analysis was not conducted and some of their assumptions seem questionable. The multinomial distribution was also used by Unser (1986) to model the components of his sum and difference histograms. Whilst these methods avoid the use of arbitrary and lossy features, and have the advantage of statistical techniques, they are dependent on the appropriateness of the assumed stochastic model.

3.2.4 Local Texture Properties

Representing texture by global statistics is appropriate during classification because each image contains a homogeneous texture sample, but local properties may be more reliable because they are less affected by smooth changes arising from the image acquisition environment. Local features are also preferred during segmentation, when the regions over which statistics must be collected may be small or irregular (§3.1.2). Exploitation of local image properties is consistent with texton theory (Julesz, 1981), which marks texture boundaries where there is a difference in first-order feature statistics (§2.2.1). Compactness and spatial localisation are important characteristics of primitive features apparently employed by the human visual system, and our justification for the observed dual-paradigm structure requires that they may be extracted by simple data-parallel processing (§2.3). Data-independent transformations are discussed in this section: their objective is to allow texture edges to be located in the filtered image by “standard” first-order techniques (Figure 1.3 on page 5). Methods operating at a slightly higher level are discussed below (§3.3.3).

A system for segmenting images by thresholding the value of a local image property was described by Zucker, Rosenfeld and Davis (1975). This relied on a global histogram of the transformed image being bimodal when two texture types were present, one peak corresponding to each texture. Image statistics were extracted after convolution with “spot detector” kernels, whose spot size was adjusted automatically, but non-maximal suppression was required to form clear peaks in the histogram. Such a simple operator has only limited discriminating power but could be extended by a larger set of local properties (Zucker *et al.*, 1975).

Purely linear filtering is invertible and hence cannot extract information, but linear convolution is computationally attractive and a common technique is to compose it with a point non-linearity such as squaring or absolute value (Caelli, 1985). Linear averaging may be used to overcome spatial variability and “noise” in feature output, but this can incorrectly eliminate small regions and mislocate boundaries, and non-linear methods may perform better (§3.5.3). Particular care is required in the vicinity of texture boundaries because the region of support of the feature detector may span more than one texture type, with unpredictable results, but techniques derived from edge-preserving smoothing may be employed to reduce errors of this type (Hsiao & Sawchuk, 1989).

Texture Energy Measures

Many filter kernels have been proposed for use in conjunction with Caelli's (1985) framework. A set of standard masks, designed to respond strongly to heuristic spatial features such as bars and rings, was proposed by Laws (Ballard & Brown, 1982; Hsiao & Sawchuk, 1989; Pietikäinen *et al.*, 1983), and similar kernels were proposed as edge or spot detectors (Caelli, 1988; Zucker *et al.*, 1975). Precise numerical values appear less important than the kernel's general form (Pietikäinen *et al.*, 1983). Indirectly, these filters measure the local spatial frequencies present in an image, discarding phase, and are known as *texture energy measures*. Empirical classification results with Brodatz textures suggest that Laws' kernels achieve higher accuracy than second-order co-occurrence statistics (Pietikäinen *et al.*, 1983).

Spatial/spatial-frequency dependence may be made more explicit by the use of the Wigner representation (Reed & Wechsler, 1990), effectively a spatially-restricted Fourier transform. Recently, tunable Gabor kernels have attracted much interest because they achieve optimal joint resolution in the spatial and spectral domains (Daugman, 1985; Gabor, 1946). This theoretical advantage is reinforced by empirical studies which have found Gabor kernels suitable for distinguishing many natural and synthetic textures (Bovik *et al.*, 1990; Tan, 1988). We discuss the attractive properties of these filters in more detail in Chapter 4, then review some previous applications to image analysis before describing our experiments and proposals for novel Gabor texture features.

Texture energy measures may be computed quickly and easily, and do not require data-dependent processing. These characteristics are similar to those observed for pre-attentive human vision, and imply analogous computational benefits (§2.4). Unlike features collected from global histograms, the simplicity of energy measures allows the formation of complete filtered images, assigning a feature value to each site, and this is particularly advantageous during segmentation (§3.1.2). Features are robust to smooth changes in texture parameters because they are collected locally, and sophisticated feature-reduction methods have been proposed to identify which members of the filter family contribute most strongly to texture discrimination (Caelli, 1988; Unser & Eden, 1989).

3.2.5 Random Field Models

None of the texture models described in this Chapter is in perfect agreement with real image data. Even disregarding variability arising from external sources, such as surface geometry, lighting, and camera parameters, correspondence between abstract models and real textures is inexact. Disparities arise because the representations we have discussed ignore an important and fundamental characteristic of natural texture: its *variability* (§1.3). Whilst random variation can never be predicted exactly, it may be described, analysed and *expected* behaviour predicted by stochastic modelling, which forms probabilistic descriptions. This framework captures complex random variation in a small number of parameters, and allows models to be manipulated by optimal maximum-likelihood techniques (Devijver & Kittler, 1982). Random field models may

be used to describe both microtexture and macrotexture, but we shall postpone discussion of the latter application until §3.3.3 although the distinction between these forms sometimes vanishes.

Random field models describe the image by a multivariate probability distribution, specifying the marginal distribution for each pixel. Within a chosen family of distribution functions, each texture class may be described by selecting an appropriate set of parameter values, which determine pixel mean, variance, and correlation structure. Optimal maximum-likelihood techniques operating on the joint likelihood of all image pixels may be used to generate parameter estimates, classify textures, and segment images. Ergodicity is often assumed, implying that similar stochastic behaviour is observed within and between texture samples, although this sometimes breaks down (Gagalowicz & Ma, 1985). Once appropriate feature vectors have been obtained, they may be operated on in the same way as other derived statistics (Khotanzad & Chen, 1989; Manjunath & Chellappa, 1991). Provided that real textures vary in a similar manner to the parameterised random field models, this approach has the advantage that no loss of information occurs by the use of heuristic features. Ideally, the joint distribution would be estimated without the need to assume a parametric form, but phenomenal amounts of training data would be required to form accurate estimates able to operate at a sufficiently abstract level.

Simultaneous Autoregressive and Gibbs-Markov Models

Motivated by the original Julesz conjecture (§2.3.2), Pratt, Faugeras and Gagalowicz (1978) devised a method to synthesise random fields with specified second-order properties. A field of white noise $\mathbf{v} = \{v_i \mid i \in \mathcal{L}\}$ was convolved with a local kernel, to synthesise field \mathbf{x} :

$$\mathbf{x}_i = \mu_i + \sum_j \beta_{j-i}(\mathbf{x}_j - \mu_j) + v_i, \quad \forall i \in \mathcal{L} \quad (3.4)$$

where β is the convolution kernel and μ the pixel mean. Fields of this type are known as *simultaneous autoregressive* (SAR), and are convenient to manipulate because of their local linear neighbourhood structure (Besag, 1974). Parameter estimation may be achieved by computing a “whitening transform” that transforms \mathbf{x} back to \mathbf{v} (Faugeras & Pratt, 1980).

Autoregressive models are computationally simple, but fail to capture the full structure of natural textures. Second-order statistics of SAR fields may be modelled exactly by Gibbs-Markov models, and correspondence is exact for multivariate normal distributions. The converse is not true, and Gibbs-Markov models effectively form a superset of SAR models having greater flexibility to model natural textures (Chellappa & Kashyap, 1982). Gibbs-Markov random fields are popular texture models, and are the subject of Chapter 5. They are defined by conditional probability distributions operating on a local neighbourhood, making explicit the role of image context, and are most suitable for modelling amorphous textures not possessing long-range structure.

Random field models capture the structure of a wide variety of natural textures in a concise parameter set, the success of which may be verified conveniently by synthesis (Figure 1.4 on page 7; Khotanzad & Kashyap, 1987). Computational difficulties

often dictate the use of simplified models, such as the auto-normal Gibbs–Markov random field (§5.1.5). One general criticism of global statistics is their inability to reflect smooth changes in texture parameters, but non-stationary random field models address this problem (Silverman & Cooper, 1988). A more serious concern is that random field models do not operate at a sufficiently abstract level: they effectively perform stochastic template-matching on the image array, and accordingly are sensitive to any distortion of the texture. Image structure, of the type discussed in §3.3, cannot easily be incorporated into pixel-based random field models. Our proposals for hybrid Gabor–Markov models overcome many of these current limitations (§5.7).

3.3 Structural Approaches to Texture Analysis

Our definition of texture (page 34) offered a description at two levels, in terms of: (i) the spatial arrangement of primitive elements; and (ii) the properties they give rise to. Rather than representing textures indirectly by associated statistics, structural models seek to represent their hierarchical organisation directly (Haralick, 1979). Statistical models refer to the arrangement of pixels to form a *microtexture*; structural models to the arrangement of texture primitives to form a *macrotexture* (§1.2).

Regular textures may be described mathematically by grammars which specify tessellations of the plane (Ballard & Brown, 1982), but texture primitives must be of regular size and shape, and have a precise spatial arrangement for this model to be accurate. All equivalent texture elements possess a common property but it is not specified by the structural model — it is often taken to be a region of near-constant intensity, but could be any of the microtextures described above (§3.2). “Strong” texture models are rigid, and do not permit spatial interaction between microtextures, but only synthetic textures are perfectly regular, and inflexible deterministic models of this type are seldom used (Haralick, 1979). We shall discuss “weak” texture models, which do allow probabilistic spatial interaction (Haralick, 1979). Note that when a texture primitive consists of a single pixel, structural and statistical models are virtually equivalent (Ahuja & Rosenfeld, 1981).

Ideally, structural models are estimated by segmenting individual texture elements and noting their spatial arrangement and interaction (§1.2). This is similar to the method proposed by Marr (1976), and suggests a strong link between estimation of structural texture features and image segmentation (§3.5). In practice, this approach may be ill-specified for natural textures because the variability of both microtexture and macrotexture means there is no clear distinction between them (Tamura *et al.*, 1978). For computational reasons, it is preferable to manipulate a single texture rather than a collection of elements, and hence structural models are most appropriate to perform fine discrimination on regions located by a “pre-attentive” statistical analysis, or when the statistical approach has failed altogether.

Structural analysis is feasible for regular textures, such as straw (#15) and reptile skin (#22; Figure 1.6), because these decompose unambiguously into *micro* and *macro*-texture. Weak models of these particular Brodatz textures were formed by extracting

and grouping regions of near-homogeneous intensity (Tomita *et al.*, 1982). Several primitive texture types were identified and characterised by a number of simple parameters, such as brightness, size, and repeat vector (including level of variability), determined by examining global histograms. Reconstruction of the texture was possible using interacting placement rules, and classification was achieved by comparing feature vectors.

A weakness of this approach was the use of global histograms to determine texture behaviour, because this implicitly assumes that training images are homogeneous and not subject to perspective or other distortion, whereas both assumptions hold only under ideal conditions. This limitation may be overcome by a purely local approach (Hainey & Kanade, 1989), or by assuming particular scene geometry to guide the selection of texture primitives (Blostein & Ahuja, 1987). Systematic variations in intensity across an image are finessed by the use of more abstract feature-derived primitives, such as edges (Vilnrotter *et al.*, 1986). One study characterised Brodatz textures by average separation between oriented edge segments and claimed superior accuracy to co-occurrence or texture energy methods (Kjell & Dyer, 1985). Another took intensity extrema as primitives and achieved similar classification accuracy to co-occurrence methods, but with much less computational effort (Mitchell *et al.*, 1977).

3.3.1 Generalised Co-Occurrence Matrices

Spatial arrangements of texture primitives may be subjected to a similar second-order co-occurrence analysis to that employed above for pixels (§3.2.3), formalised by the use of *generalised* co-occurrence matrices (Davis *et al.*, 1979). Direct representation of relative locations of features of a given value is not advantageous because the resulting matrix is very sparse and fails to capture the full spatial properties of the texture, and hence indirect spatial constraint predicates are preferred. Heuristic features similar to those proposed above (3.3) were defined on these generalised matrices, and an empirical study suggested they could lead to more accurate classification than gray-level co-occurrence matrices (Davis *et al.*, 1981). Feature-based representations have the advantage of greater abstraction, and hence may be more immune to irrelevant intensity variation.

3.3.2 Statistics of Local Image Regions or “Blobs”

Methods described above make explicit the spatial arrangement of texture primitives, but structural models may also make use of other properties. The texton theory of pre-attentive human vision predicts that textures may be distinguished on the basis of differences in first-order “blob” statistics (§2.2.1), and was cited by Voorhees as the motivation for his texture segmentation program (Voorhees & Poggio, 1987). Small blobs were detected by Laplacian of Gaussian filtering (Marr & Hildreth, 1980), simple statistics computed from them, and image boundaries marked between regions whose blob attributes differed. Performance of Voorhees’ algorithm, which was not plausible as a biological model, was demonstrated only for favourable images (Voorhees, 1987; Voorhees & Poggio, 1987). An interesting extension of this approach is to characterise

textures by their fractal dimension, measured by observing the rate of variation of blob attributes as a function of spatial resolution (Peleg *et al.*, 1984).

3.3.3 Random Mosaic Models

Random mosaic models tessellate the plane nondeterministically to represent texture structure, and share many of the advantages noted above for stochastic microtexture models (§3.2.5). An additional factor is their ability to describe the physical generating process directly in some cases, obtaining a deeper correspondence with the texture origin (Ahuja & Rosenfeld, 1981). Mosaic models may be divided into two categories: (i) cell structure models tessellate the plane by defining boundary lines; (ii) coverage models by randomly distributing particular geometric shapes. Texture primitives interact according to an assumed probability distribution, for which parameters may be estimated (Ahuja & Rosenfeld, 1981; Modestino *et al.*, 1981).

As with other structural models, suitable microtextures must also be estimated to describe texture primitives, but regions of constant gray level were found inadequate to model Brodatz textures (Modestino *et al.*, 1981). More realism may be obtained by using a random field to define both macrotexture and microtexture, combining the advantages of stochastic models at two levels (Cohen & Cooper, 1987; Derin & Elliott, 1987). Structural random fields may be used to impose spatial coherence heuristics, and Gibbs random fields are particularly suitable for this purpose (§5.1.3). Hierarchical random field models are discussed in §5.6.2 — one significant drawback is their computational appetite.

3.4 Texture Classification Algorithms

Texture classification involves selecting a class label that best describes an unidentified “test” image by comparing it with sets of known “training” images (§3.1.1). Direct comparison is unwieldy and insufficiently abstract, therefore training images or texture classes are represented indirectly by feature vectors. Images are usually assumed to contain homogeneous texture samples, even though this may not be true, and a single feature vector describes the entire image. It is possible to compare the test image with class features directly, but it is often preferable to represent it by a “test” feature vector and perform indirect comparison. Errors will, of course, result if the images are not homogeneous and feature vectors are insufficiently abstract to finesse this (§3.1.3), or if the choice of texture features is inappropriate.

Which classification strategy should be followed depends in part on how much information about the feature distribution is available. Bayes classification is optimal when the joint probability density of image features is known, or a suitable parametric form may be assumed (§3.4.1); but often this is not the case, and simplified classifiers are more appropriate. We use Bayesian classification to compare the performance of our proposed Sampled-Markov and Gabor-Markov frameworks with a conventional texture analysis (Chapter 6).

3.4.1 Optimal Bayes Classifiers

Bayes classification selects the optimal class label by maximum likelihood techniques. Let us assume that we have estimated feature sets corresponding to C different texture classes, and have to decide to which class a previously unseen trial texture \mathbf{x} belongs. We assume that the *a priori* class probabilities π_i for the classes $\omega_1, \dots, \omega_C$ are known, $\sum_i \pi_i = 1$, and that all mis-classification errors incur the same penalty. We shall suspend judgement on the trial texture if it doesn't appear to fit easily into any of the classes by adopting a rejection threshold λ (Devijver & Kittler, 1982). The Bayes classifier $\hat{\omega}$ forms the class *a posteriori* probability $\mathcal{P}\{\omega_j | \mathbf{x}\}$:

$$\mathcal{P}\{\omega_j | \mathbf{x}\} = \frac{\mathcal{P}\{\mathbf{x} | \omega_j\} \pi_j}{\mathcal{P}\{\mathbf{x}\}} \quad (3-5)$$

where $\mathcal{P}\{\mathbf{x}\}$ is the unconditional density of \mathbf{x} (effectively a normalising constant). We select the class with highest probability:

$$\begin{aligned} \Pi &= \max_j \mathcal{P}\{\omega_j | \mathbf{x}\} \\ \hat{\omega}(\mathbf{x}) &= \begin{cases} \omega_i, & \mathcal{P}\{\omega_i | \mathbf{x}\} = \Pi \geq 1 - \lambda \\ \omega_0, & 1 - \lambda > \Pi \end{cases} \end{aligned} \quad (3-6)$$

forming the *maximum a posteriori* (MAP) class estimate $\hat{\omega}$. If there is a tie between two or more classes, any one of them is chosen. When $\lambda \geq \frac{C-1}{C}$, the reject option ω_0 is unavailable.

Random field models assume that image pixels follow a particular distribution parameterised by the class vectors, and in this context a lossy intermediate representation may be avoided by representing the test image by its pixel array or pattern vector, \mathbf{x} in (3-6). Classification then reduces to the simple task of evaluating *a posteriori* likelihoods.

An alternative approach is to represent the test image by another parameter vector, \mathbf{x} . Maximum likelihood methods may still be used provided the joint density of feature coefficients is known, and in practice these are often assumed to follow a particular distribution, multinomial and multivariate Gaussian distributions having been used for this purpose (Unser, 1986; Vickers & Modestino, 1982).

Classification of feature vectors is a classical theme in statistical pattern recognition, and many approaches are available (Devijver & Kittler, 1982). It is usual to view feature vectors as points in a parameter space, which is partitioned between the various texture classes during training, possibly including a reject option, so that test vectors may be assigned the correct class label quickly. In general, the nature of the class boundary or discriminant function is arbitrary, but for the special case of a two-class problem, when feature vectors are normally distributed with common covariance Σ and means μ_i , the decision boundary is a hyperplane:³

$$\mathbf{x}^T \Sigma^{-1} (\mu_1 - \mu_2) = \frac{1}{2} (\mu_1 + \mu_2)^T \Sigma^{-1} (\mu_1 - \mu_2) + \log(\pi_2/\pi_1) \quad (3-7)$$

where π_i are the *a priori* probabilities (Devijver & Kittler, 1982, page 40). Effectively, this plane defines the direction which maximally discriminates two classes, along which the test vector is projected.

³This special case is known as the Anderson discriminant plane.

3.4.2 Probabilistic Distance Measures

Our Bayesian classification rule (3.6) does not suggest how reliably two classes may be discriminated, and Bayes error cannot usually be evaluated exactly because it requires a complicated integration over the entire feature space, \mathcal{S} . More tractable probabilistic distance measures may be used to bound it, of which two are of particular interest: the *Bhattacharyya* coefficient J_B , and *Mahalanobis* distance Δ_M .

$$J_B = -\log \int_{\mathcal{S}} \sqrt{\mathcal{P}\{x | \omega_1\} \mathcal{P}\{x | \omega_2\}} dx \quad (3-8)$$

$$\Delta_M = (\mu_1 - \mu_2)^\top (\pi_1 \Sigma_1 + \pi_2 \Sigma_2)^{-1} (\mu_1 - \mu_2) \quad (3-9)$$

The *Bhattacharyya* coefficient J_B gives a better estimate for the Bayes error than the *Mahalanobis* distance, but cannot be evaluated except in special cases; for example, when feature vectors are drawn from a multivariate normal distribution $G(\mu_i, \Sigma_i)$ (Faugeras & Pratt, 1980):

$$J_B = \frac{1}{4} (\mu_1 - \mu_2)^\top (\Sigma_1 + \Sigma_2)^{-1} (\mu_1 - \mu_2) + \frac{1}{2} \log \left(\frac{|\frac{1}{2}(\Sigma_1 + \Sigma_2)|}{\sqrt{|\Sigma_1| |\Sigma_2|}} \right). \quad (3-10)$$

In the case of equal covariance matrices and *a priori* probabilities, $\Sigma_1 = \Sigma_2$ and $\pi_1 = \pi_2$, the *Bhattacharyya* and *Mahalanobis* measures are equal. Probabilistic distance measures indicate the distinctiveness of pairs of texture classes, and are useful to assess the performance of feature sets independently of subjective criteria or classification algorithm (§3.6.1), and in determining which regions to merge during agglomerative clustering (§3.5).

3.4.3 Sub-Optimal Discriminant Functions

A linear decision boundary is optimal for multivariate normal distributions with common covariance matrix, but in practice is often applied under other circumstances as well because of its computational simplicity. For two classes with arbitrary feature distributions, we may project the test feature vector onto the direction:⁴

$$(\Sigma_1 + \Sigma_2)^{-1} (\mu_1 - \mu_2) \quad (3-11)$$

which maximises the ratio of between-class and within-class variation (Devijver & Kittler, 1982). It reduces to the maximum-likelihood discriminant plane when the above conditions are satisfied, and may be extended to multiclass discrimination by computing:

$$d_i = (\mu_i - x)^\top \Sigma_i^{-1} (\mu_i - x)$$

and maximising d_i with respect to the class label i (Weszka *et al.*, 1976).

In practice, there may be insufficient feature measurements defining each class to estimate covariance matrices reliably, and a further approximation may be taken by assuming that they are diagonal (Chellappa & Chatterjee, 1985; Kashyap & Khotanzad,

⁴Known as the Fisher linear discriminant.

1986). This implies features are independent, which is unrealistic, but does allow the variability of each feature to be taken into account (Coleman & Andrews, 1979). Similarity with (3-10) has led some researchers to refer optimistically to this *normalised* Euclidean distance as the Bhattacharyya distance (Chellappa & Chatterjee, 1985).

When even the diagonal elements of the covariance matrix are unavailable, perhaps because each class is defined by a single vector, there is limited scope for statistical rigour. One popular measure is the *weighted* Euclidean distance (Manjunnath & Chellappa, 1991):

$$\Delta_w = (\mathbf{x} - \mathbf{y})^T \mathbf{W}^{-1} (\mathbf{x} - \mathbf{y}) \quad (3.12)$$

where $W_{k,\ell} = (x_k^2 + y_k^2)\delta_{k,\ell}$, but this is sensitive to a translation of feature origin. Alternatively, the Euclidean distance may be used directly, following the nearest-neighbour rule (Devijver & Kittler, 1982; Mitchell *et al.*, 1977). This form is dimensionally inhomogeneous, and should be considered as a last resort.

3.4.4 Optimal Feature Selection

We have assumed above that the logically distinct operations of feature extraction and classification are indeed performed separately, but this need not be the case. When feature extraction or model matching carries a high computational cost, it is advantageous to structure the classification as a decision tree (Tomita *et al.*, 1982; Vilnrotter *et al.*, 1986). Features with greatest discriminating power are placed near the root of the tree, and direct subsequent processing accordingly.

Similar principles guide the selection of an optimal feature set. Feature vectors are sometimes generated by an entire family of filters, many members of which have little resolving power. A feature extractor matrix transforms the current feature set into a smaller vector whilst optimising a probabilistic distance criterion (Devijver & Kittler, 1982). Compression algorithms may be used to identify the most significant components within each region, but these do not always pick good discriminants (Unser & Eden, 1989). Orthogonal features may be selected by comparison of “one-at-a-time” Bhattacharyya coefficients (Coleman & Andrews, 1979; du Buf *et al.*, 1990).

3.5 Image Segmentation Algorithms

Segmentation algorithms partition an image into disjoint regions, each homogeneous with respect to some property. Automatic segmentation algorithms do not require prior knowledge of the number or type of textured regions, but estimate these from the scene without external intervention. Additional constraints must be provided to specify segmentation fully, either in the form of heuristics reflecting the continuity and spatial coherence of physical objects, or a parameterised image model. Similar constraints are suggested by models of human vision (Chapter 2), and seek to express expectation about the nature of the physical world (Marr, 1982; Tsotsos, 1987): boundaries are smooth and continuous, and regions do not contain many holes or small pockets. Heuristics may drive the segmentation from the start, or can be applied during post-processing (§3.5.3).

Smoothness and homogeneity constraints apply principally to the underlying physical object and may not be fully reflected in the image. When the image is modelled by random fields, maximum-likelihood segmentation may be achieved by extremising a cost function.

Segmentation requires analysis of both spatial and statistical properties, leading to a conflict because image features are estimated accurately from large areas but spatial properties from small regions (Wilson & Granlund, 1984). Even when all texture parameters are known *a priori*, segmentation is an extremely difficult problem because hypothetical boundaries increase exponentially with the number of parameters, and exhaustive search is infeasible (Gurari & Wechsler, 1982). Approximate solutions may be obtained by directed local search, fully evaluating only a small proportion of possible segmentation hypotheses. Following the central dichotomy of low-level human vision (Chapter 2), this may take the form of a two-stage process: "pre-attentively" defining a region of interest to establish the vicinity of the border; and "attentive" refinement to locate it accurately (Blanz & Reinhardt, 1981). It may be more efficient to use a process of continual refinement rather than a two-stage algorithm, supported by multi-resolution representations (Sber & Rosenfeld, 1989; Spann & Wilson, 1985). Experimental segmentation of synthetic textures and Brodatz montages using Gabor filtering is described in Chapter 4.

3.5.1 Merging and Splitting Image Regions

The two main approaches to establishing distinct image regions are merging and splitting (Ballard & Brown, 1982). If texture parameters are unknown *a priori*, they may be estimated reliably from large regions of a single texture type — but these are only available after the segmentation is complete! A compromise is to impose a grid structure on the image, and to estimate parameters within each of the regions it defines. The size of these regions should be sufficient for parameters to be estimated with some confidence but small enough for most regions to contain a single texture type. Adjacent regions with compatible parameters are merged, and their parameters re-estimated, making explicit the application of both spatial and statistical constraints. Agglomerative clustering continues until: a given number of regions remain (Coggins & Jain, 1985), an arbitrary threshold is reached (Caelli, 1985), or according to a statistical criterion (Silverman & Cooper, 1988); but the resulting border still reflects the imposed grid structure, and defines the *approximate* location of the true boundary. Small errors may occur because flawed merging decisions may not be reversed by this algorithm, and some initial regions may straddle texture boundaries, but segmentation of non-stationary images is facilitated by using local properties.

Splitting algorithms proceed in the opposite direction, from coarse to fine resolution, and recursively decompose image regions. Purely statistical approaches apply a threshold derived from a bimodal feature histogram to segment regions (Zucker *et al.*, 1975), but per-pixel classification does not take full account of spatial context, leading to ragged boundaries and clumps of isolated pixels. Spatial coherence may be imposed by a post-processing step (Caelli, 1985).

Spatial constraints may be employed directly by computing features over image regions, leading to more reliable feature estimates. A quad-tree structure (Spann & Wilson, 1985) or resolution pyramid (Burt & Adelson, 1983) is useful to represent the image at several levels of spatial resolution, and may contain image or feature information (Sher & Rosenfeld, 1989). Nodes near the apex project to many “children” near the base, and a coordinate grid is imposed within each layer. Per-pixel classification near the top of the pyramid defines approximate locations of texture boundaries, and these estimates are refined at finer resolution by reclassifying pixels in their vicinity, terminating at the base of the pyramid. As with region merging, the boundary is approximate: errors cannot be recovered from, and the imposed grid structure may lead to fragmentation; but this is addressed by a combined split-and-merge algorithm (Ballard & Brown, 1982; Chen & Pavlidis, 1983). Defining approximate image boundaries at each resolution selectively targets expensive local search to key image locations, achieving accurate segmentation with high efficiency.

3.5.2 Detection of Texture Edges

Splitting and merging algorithms apply to image regions, and are suitable when texture features may only be computed at grid points; for example, it is inefficient to extract a feature vector for each image location when they are derived from histograms or co-occurrence matrices (§3.2.3). Other features, such as texture energy measures, may readily be computed locally, and allow the use of first-order edge-detectors to locate image boundaries (Figure 1.3 on page 5). Texture-edge detectors respond only to local changes, and offer greater robustness when image boundaries are confounded by noise and smooth texture variation (Nahi & Jahanshahi, 1977). Another advantage is that small but distinct regions possess clear image boundaries, but may fail to generate distinguishable peaks in global histograms and hence be overlooked by region-based algorithms (Bhanu & Parvin, 1987). Ideally, combined edge and region processes should be used (Geman & Geman, 1984).

Multiple resolution representations are useful in refining the location of image borders efficiently. Putative texture edges are first detected near the apex of the pyramid, and are located conservatively: the width of the border region is a constant number of pixels, but projects into a smaller area in the image at successively finer resolution (Spann & Wilson, 1985). Pyramidal processing is driven by the principle that expensive high-resolution processing is guided by approximate results computed at lower resolution, in accord with the Guided Search model of pre-attentive human vision (§2.4.1). A clear dichotomy between serial and parallel processing does not emerge because there is a graded distinction between “coarse” and “fine” detail.

3.5.3 Refinement of Approximate Texture Boundaries

Region merging leads to jaggy texture boundaries, reflecting the imposed coordinate structure; feature clustering to ragged and noisy boundaries because spatial continuity constraints have not been fully expressed; and texture edges may be fragmented or incomplete, forming a partial segmentation. In each case, acceptable segmentation

accuracy may be achieved by carefully adjusting boundary position according to selected optimisation criteria. It is often acceptable to restrict analysis to local changes of the existing approximate segmentation boundaries, perhaps by using dynamic programming (Derin & Elliott, 1987).

Relaxation labelling processes impose many local constraints simultaneously in order to achieve a globally consistent scene interpretation (Kittler & Illingworth, 1985). Each node communicates only with its local neighbours, adjusting its state according to local criteria, but constraints propagate over the network allowing an acceptable sub-optimal solution to be found. Relaxation algorithms are inherently parallel and are a popular model for simultaneous expression of many local constraints (Caelli, 1985), but careful control is required in order to ensure termination whilst allowing many avenues to be explored. "Greedy" update rules, which always select the optimal *local* state, ensure convergence but do not explore the space of solutions very thoroughly. Relaxation algorithms benefit from multiple-resolution representations because the rate of convergence is increased (Kittler & Illingworth, 1985; Terzopoulos, 1986). When the objective is to refine an approximate initial boundary, greedy optimisation at successively finer resolution is an efficient approach (Bouman & Liu, 1991).

Relaxation algorithms may be used in conjunction with recursive splitting, magnifying local image contrast to encourage the global feature histogram to become bimodal (Bhanu & Parvin, 1987), and are influenced by spatial organisation without requiring arbitrary partitioning of the image. Added emphasis on local context diminishes the influence of smooth variations of texture properties across the image. Probabilistic relaxation allows a wider range of possible solutions to be explored but often at considerably increased computational cost (Geman & Geman, 1984; Hsiao & Sawchuk, 1989).

3.6 Discussion of Current Approaches

Many of the approaches to texture manipulation discussed in this Chapter arose from heuristic proposals rather than dispassionate analysis of information-processing requirements. Before outlining our approach to the development of improved texture representations (§3.6.2), we first discuss how competing models may be evaluated.

3.6.1 Evaluation of Feature-Extraction Methods

Comparison between feature extraction methods is desirable in order to discover which feature set performs best in what circumstances, and why. A number of approaches have been used, with varying degrees of objectivity and generality. By far the most commonly used protocol is "classification result comparison", in which known texture samples are classified by a single algorithm with different feature sets and performance is determined quantitatively by the proportion of textures identified correctly (Connors & Harlow, 1980). This methodology has been used to support many claims of the superiority of one feature extraction method over another, often using textures from the Brodatz album (*e.g.* Pietikäinen *et al.*, 1983; Unser, 1986; Weszka *et al.*, 1976). We adopt this approach to compare the performance of our proposed Sampled-Markov

and Gabor–Markov frameworks with a conventional Markovian analysis using optimal Bayesian classification (Chapter 6).

Empirical classification accuracy is straightforward to measure and gives a quantitative comparison, but suffers from a number of drawbacks. Purely as a performance measure, it is not independent of the type of images used or the classification algorithm itself. In order to have high confidence in the verdict as a general result, statistics would have to be collected for a huge variety of images types (medical, aerial, thermal, outdoor scenes, noise-corrupted images, etc.), using several classification algorithms. A further drawback of empirical classification accuracy is that it fails to identify *why* one feature set performs better than another.

Synthesis of an artificial image from measured feature sets followed by visual comparison with the original is effective in checking that major texture characteristics have been captured, and this method has been used to guide feature selection (Khotanzad & Kashyap, 1987). Knowledge of some statistical properties, such as those obtained from co-occurrence matrices, does not suggest a manner by which a texture may be synthesised. Although it is often straightforward to generate artificial textures possessing the layout characteristics of structural models, it may still be impossible to synthesise their microtexture component if this is defined by statistical properties (Ahuja & Rosenfeld, 1981; Tomita *et al.*, 1982). In one case, comparison with the original texture prompted an admission that the proposed model was inadequate (Modestino *et al.*, 1981). One great advantage of random field models is that synthesis is always possible, allowing the influence of parameter sets and extraction methods to be compared easily (Cross & Jain, 1983). Artificial textures are readily synthesised from Gibbs–Markov models using techniques discussed in Chapter 5 (Figure 1.4 on page 7), and further examples are presented in Chapter 6.

A more objective evaluation of feature performance is obtained by determining the ideal classification error with a given set of feature measurements, independent of classification algorithm. Bayes error cannot easily be computed exactly but is closely bound by the Bhattacharyya distance (§3.4.2), which has been used for this purpose although it is still tied to a particular set of test data (Coleman & Andrews, 1979; Fangeras & Pratt, 1980). Another restriction is that it is necessary in practice to assume that the feature vector is multivariate Gaussian in order for evaluation of the Bhattacharyya coefficient to be feasible, even when this is clearly false (Davis *et al.*, 1981).

Recently, it has been argued that global statistics of this type are inappropriate criteria when features are to be used for image segmentation because contradictions may arise between statistical and spatial accuracy (du Buf *et al.*, 1990; Wilson & Granlund, 1984). Disagreement also exists over whether comparisons should be made with natural or synthetic textures: the former have the advantage of natural variability rather than following a prescribed model; but image parameters may be more closely controlled with synthetic image textures.

A further improvement of relevance to practical systems would be to assess performance in conjunction with the amount of computation required. For a given abstract architecture, for example, performance of different feature extraction and processing algorithms could be compared as a function of permitted CPU time. Features which

may be processed by data-parallel machines are potentially much more efficient than those requiring local data-dependent analysis (§A.1).

3.6.2 Summary: Major Aspects of Current Texture Models

Approaches to texture analysis are commonly divided into statistical and structural models, analogous with the attentive dichotomy in low-level human vision (Chapter 2). Structural analysis seeks to identify individual texture elements, describing the image in terms of their properties and spatial arrangement (§3.3). Although texture hierarchy and generating processes may be represented explicitly, structural approaches are often computationally demanding, and do not fit the data-parallel paradigm (§A.1).

By contrast, statistical approaches do not attempt to form complete image representations, but describe textures indirectly by simple derived properties or features which may be extracted and manipulated efficiently, and require little data-dependent processing (§3.2). Texture features are often computed from second-order image statistics, then first-order feature differences used to locate hypothetical texture boundaries, consistent with models of pre-attentive human vision (§2.2). Structural models have been less thoroughly studied, perhaps because of their requirement for greater computational resources, but form a superset of statistical representations and hence are potentially superior (Ahuja & Rosenfeld, 1981). Random field models explicitly describe texture variability, and hence are suitable for representing irregular natural textures, but hierarchical random field models are very computationally demanding.

Our objective is to combine the computational efficiency and convenience of statistical models with the superior descriptive power of structural approaches, while acknowledging random texture variability. We have identified Gabor filtering as an attractive means of extracting statistical features, and Gibbs-Markov random fields as particularly appropriate for expressing the probabilistic influence of spatial context, and we examine these paradigms in more detail in Chapters 4 and 5, motivating our proposed Gabor-Markov framework for texture analysis.

4

Gabor Segmentation Experiments and Feature Extraction Algorithms

Useful physical information from several sources is bound up in the gray-level image, and early vision algorithms seek to recover it by making heuristic assumptions about the spatial properties of the scene. Segmenting the image into different textured regions is an important component of the transition from image to object properties, and many solutions to this problem have been proposed. Our review of current approaches to texture classification and segmentation in Chapter 3 identified texture energy filtering as a promising technique, offering good performance in empirical comparative studies but amenable to efficient parallel implementation because processing is largely data-independent (§A.1). The texture energy algorithms we discussed previously employ heuristic filter kernels, designed for practical rather than theoretical convenience. Examination of physical objects requires simultaneous analysis of spatial and spectral properties, which Gabor filters uniquely achieve with minimum uncertainty (§4.1), and therefore we anticipate that the performance of texture segmentation algorithms using Gabor filters should be comparable or superior to those reported previously. Gabor filtering has also been suggested as a mechanism to support perceptual theories of human vision, motivated partly by experimental physiological data (Chapter 2). In this Chapter, we study Gabor filters in greater detail, and suggest how they may be exploited in a novel framework for texture analysis.

Previous applications, reviewed briefly in §4.3, have concentrated both on the performance of Gabor filters in their own right and as models of human visual processing.

Building on these results, our experiments with natural and synthetic textures confirm that simple algorithms exploiting Gabor energy are able to segment suitably-constrained images, offering levels of performance qualitatively similar to those of pre-attentive human vision, and demonstrate the potential of Gabor filtering for image analysis. The two major difficulties with this approach are: firstly, residual variability within each textured region after Gabor filtering, which interferes with attempts to extract valid image boundaries and must be suppressed by post-processing; and secondly, the lack of a principled strategy to combine responses from many Gabor channels, most of which contribute little useful information to the segmentation process.

Our approach is to describe texture by hierarchical models which exploit rather than suppress fluctuations in Gabor channels, describing the spatial arrangement of Gabor features by random fields (Chapter 5). Observed response spectra are matched against parameterised Gabor signatures by efficient optimisation algorithms in a novel feature-extraction framework (§4.6), producing a sensitive and compact representation suitable for subsequent modelling. We adopt two approaches to the difficult task of feature estimation, and demonstrate that these generate acceptable feature vectors from real textures (§4.6.4). Suitable random field models are discussed in Chapter 5, and we apply our novel Gabor-Markov models to the analysis of real textures in Chapter 6.

4.1 Theoretical Properties of Gabor Filters

Gabor originally proposed the class of filters which bears his name in the context of communication theory (Gabor, 1946). He had become convinced that the maximum amount of information which a signal could convey was proportional to both its duration and frequency bandwidth. Gabor proposed a “diagram of information”, with time and temporal frequency axes, in which quanta of information were represented by cells. As a communications engineer, his goal was to identify the elementary signals or “logons” which could represent information most efficiently with the smallest possible area. A fundamental conflict between simultaneous localisation along both dimensions, predicted by analogy with quantum theory, prevents this area being reduced to a point. A signal located exactly in the frequency domain, a sinusoid, simultaneously occupies the entire time axis because there is no concept of “varying frequency”, and similarly a δ -function pulse occupies a single point on the temporal axis but is dispersed across all spatial frequencies. The uncertainty principle predicts the minimum area in Gabor’s diagram of information which any signal may occupy:¹

$$\Delta t \Delta f \geq \frac{1}{4\pi} \quad (4.1)$$

where Δt and Δf represent uncertainty along the temporal and temporal-frequency axes, respectively, in measurements derived from linear filters. The elementary signals sought by Gabor achieve equality in (4.1), and may be written as the product of a Gaussian envelope with a sinusoid (Gabor, 1946).

¹Gabor (1946) included a factor of $\sqrt{2\pi}$ in his definitions of Δt and Δf , reaching the equation: $\Delta t \Delta f \geq 1/2$.

The dichotomy between temporal and temporal-frequency decomposition is mirrored in image processing, where the relevant quantities are space and spatial frequency (in one or two dimensions). Fourier methods, discussed briefly in §3.2.2, seek to identify a pattern by its spectral components alone, discarding all spatial (phase) information. By contrast, the location of a single pixel is known exactly, but it does not describe spatial pattern and hardly provides sufficient information to recognise an object. As noted in §3.5, visual processing demands *simultaneous* identification and localisation of physical objects appearing in an image, but the uncertainty principle implies that this may be achieved only by compromising both the bandwidth and spatial extent of the filtering kernel (Wilson & Granlund, 1984). In this case, the motivation is to characterise signals with minimum *joint* uncertainty. In two dimensions, similar relations to (4.1) hold independently along each axis (§A.2.1):

$$\begin{aligned}\Delta x \Delta u &\geq \frac{1}{4\pi} \\ \Delta y \Delta v &\geq \frac{1}{4\pi}\end{aligned}\quad (4.2)$$

where u and v are spatial frequencies along the x and y directions, respectively (Daugman, 1985; Wilson & Granlund, 1984). These may be combined into a single relation:

$$\Delta x \Delta u \Delta y \Delta v \geq \frac{1}{16\pi^2}\quad (4.3)$$

but (4.2) always takes precedence. Relation (4.3) defines the minimum volume each filter may sample in an information hyperspace. Members of the Gabor family are unique in achieving minimum joint uncertainty, shown by equality in (4.2) and (4.3).

4.1.1 Specification of Gabor Kernels

It is convenient to represent the amplitude and phase of a signal by complex numbers. In the most general form, a 2-D Gabor kernel $q(x, y)$ of type $\mathbb{R}^2 \rightarrow \mathbb{C}$ is given by:

$$q(x, y) = \exp[-(Ax^2 + Bxy + Cy^2 + Dx + Ey + F)]\quad (4.4)$$

where $B^2 < 4AC$ and D, E and F are complex (Daugman, 1985). This function may be written as the product of an envelope with a sinusoid, and we shall restrict our attention to the case when these are aligned along a common axis because filters of this type correspond to more intuitive image features. The resulting expression is simplified by rotating the coordinate system by an angle θ to align it with this axis:

$$q_\theta(x, y) = \frac{1}{2\pi\sigma_x\sigma_y} \underbrace{\exp\left(-\frac{x'^2}{2\sigma_x^2}\right) \exp\left(-\frac{y'^2}{2\sigma_y^2}\right)}_{\text{envelope}} \underbrace{\exp(j\omega x')}_{\text{sinusoid}}\quad (4.5)$$

where²

$$x' = R_\theta x.$$

Note that θ is measured clockwise from vertical. Effectively, filtering with this kernel (4.5) performs a one-dimensional Fourier transform under a Gaussian envelope. The

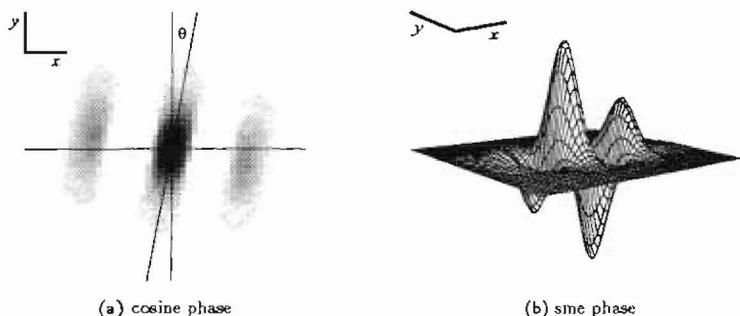


FIGURE 4.1: Typical Gabor kernel. Real and imaginary parts of the Gabor filter are in approximate quadrature. The origin lies at the centre of each plot. (a) cosine phase (real), viewed from above (only positive values are shown). (b) sine phase (imaginary). Filter parameters are: $(T = 14.2$ pixels, $B_{1/2} = 1$, $\Delta\theta_{1/2} = \pi/6$, $\theta = \pi/18$; $\sigma = 8$ pixels, $\omega = 0.44$, $\lambda = \sqrt{3}$), using the notation described on page 69.

spatial form of a Gabor kernel is shown in Figure 4.1: it is clear that the parameter θ determines the orientation to which $q_\theta(x, y)$ responds most strongly, by defining the axis along which the frequency analysis is conducted. Interaction with the Gaussian envelope means that the two phase components (real and imaginary) are no longer in perfect quadrature. Note that the Fourier and Gaussian kernels are special cases of the Gabor family, obtained by setting $\sigma \rightarrow \infty$ and $\omega = 0$ respectively. Gabor functions are unique in maintaining the joint uncertainty limit for all intermediate cases.

It is convenient to name the eccentricity of the Gaussian envelope explicitly and to recognise the cos and sin phases by writing $\lambda = \sigma_x/\sigma_y$ and $q = q_c + jq_s$:

$$\begin{aligned} q_c(\lambda, \theta, x, y) &= \frac{\lambda}{2\pi\sigma^2} \exp\left(-\frac{x'^2}{2\sigma^2}\right) \exp\left(-\frac{y'^2\lambda^2}{2\sigma^2}\right) \cos(\omega x') \\ q_s(\lambda, \theta, x, y) &= \frac{\lambda}{2\pi\sigma^2} \exp\left(-\frac{x'^2}{2\sigma^2}\right) \exp\left(-\frac{y'^2\lambda^2}{2\sigma^2}\right) \sin(\omega x') \end{aligned} \quad (4.6)$$

where $\sigma = \sigma_x$.

The frequency response $Q_\theta(u, v)$ of the Gabor filter $q_\theta(x, y)$ is also expressed conveniently in rotated coordinates:

$$Q_\theta(u, v) = \exp\left[-\frac{1}{2}(2\pi u' - \omega)^2 \sigma^2\right] \exp\left[-\frac{1}{2}(2\pi v' \sigma / \lambda)^2\right] \quad (4.7)$$

where $u' = \mathbf{R}_\theta u$. Q_θ is simply a Gaussian with aspect ratio $1/\lambda$, displaced according to the spatial frequency $u_0 = \frac{1}{2\pi} \mathbf{R}_\theta^{-1}(\omega, 0)^T$. The frequency response of a typical (complex) filter is shown in Figure 4.2 — note that it is often drawn for the separate phase components, q_c and q_s , which have lobes at $\pm u_0$.

$$\begin{aligned} q_c(\omega) &= \frac{1}{2} [q(\omega) + q(-\omega)] \\ q_s(\omega) &= \frac{1}{2} [q(\omega) - q(-\omega)] \end{aligned}$$

²The matrix \mathbf{R}_θ rotates axes clockwise by an angle θ : $\mathbf{R}_\theta = \begin{pmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{pmatrix}$.

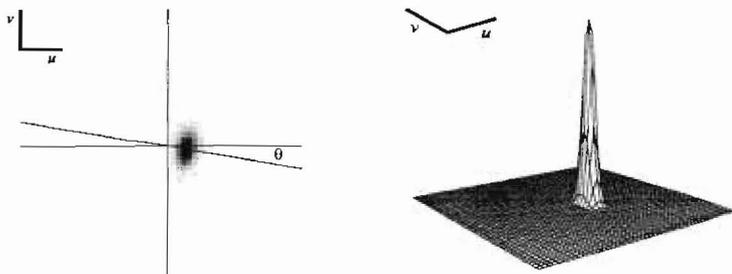


FIGURE 4.2: **Fourier transform of typical Gabor kernel.** The frequency response of a complex Gabor kernel is a Gaussian displaced according to filter frequency and orientation. The frequency origin lies at the centre of each plot. Fourier transforms of separate phase components have two lobes placed symmetrically about the origin. Filter parameters are: ($T = 14.2$ pixels, $B_{1/2} = 1$, $\Delta\theta_{1/2} = \pi/6$, $\theta = \pi/18$; $\sigma = 8$ pixels, $\omega = 0.44$, $\lambda = \sqrt{3}$), using the notation described on page 69. This filter was shown in Figure 4.1.

Symmetry of (4.2) implies that we may exchange q_θ and Q_θ , corresponding to spatial filtering with a displaced Gaussian, but this is not of practical interest.

4.1.2 Suitable Parameterisation of Filter Kernels

In the form expressed in (4.6), a Gabor kernel has four parameters: $q(\lambda, \sigma, \omega, \theta)$. Of these, only ω and θ directly reflect image properties, namely orientation and spatial frequency, and consequently alternative parameterisations are more appropriate. We may derive expressions for orientation and principle spatial-frequency bandwidths and use these to characterise the filter (§A.2.2):

$$\begin{aligned}\Delta\theta_{1/2} &= \arctan\left(\frac{\alpha\lambda}{\sigma\omega}\right) \\ \Delta\omega_{1/2} &= \alpha/\sigma \\ B_{1/2} &= \log_2\left(\frac{\sigma + \alpha}{\sigma - \alpha}\right)\end{aligned}\quad (4.8)$$

where $\alpha = \sqrt{\ln 4}$. We shall use half-peak bandwidths throughout: $\Delta\omega_{1/2}$ (radians per pixel) is the half-peak angular spatial-frequency bandwidth measured along the x' -axis; $B_{1/2}$ (dimensionless) is the same quantity but measured in frequency octaves; $\Delta\theta_{1/2}$ (radians) is the half-peak orientation bandwidth (Bovik *et al.*, 1990). Taking bandwidths as primary characteristics, we may parameterise the filter kernel as: $q(\omega, B_{1/2}, \Delta\theta_{1/2}, \theta)$ or $q(T, B_{1/2}, \Delta\theta_{1/2}, \theta)$, where $T = 2\pi/\omega$. With this form, appropriate sampling intervals of θ and ω are easily obtained, and typically we may choose: $\Delta\theta_{1/2} = \pi/6$ and $B_{1/2} = 1$; from (4.8) this gives $\lambda = \sqrt{3}$ and $\sigma\omega = 3\alpha$, as shown in Figures 4.1 and 4.2.

Parameters for continuous filters may be chosen at will, because optimal joint resolution is achieved by any member of the Gabor family, but discrete convolution is employed in practice and acceptable parameter sets are dictated by the requirement

to restrict aliasing errors. Some aliasing is inevitable because the kernel $Q_\theta(x, y)$ (4.5) is not band-limited. Since the sinusoid is always aligned with the x' -axis, we may safely assume that the critical sampling density lies in this direction, and conduct a one-dimensional analysis to determine the relative error ξ (§A.2.3):

$$\xi = \frac{\Phi[\gamma(1 - \frac{\sigma}{\omega\sqrt{2}})]}{\Phi(\gamma)} \quad (4.9)$$

where

$$\gamma = \alpha\sqrt{2} \frac{2^{B_1/2} + 1}{2^{B_1/2} - 1}$$

and $\Phi(\cdot)$ is the cumulative normal distribution. In all cases, the sampling theorem dictates that the image must be sampled at least twice within each sinusoid period (Rosenfeld & Kak, 1982). Radial anisotropy of the rectangular pixel grid forces the image to be sampled a little more densely than this along the coordinate axes, to ensure that the condition is met in all directions.

$$\begin{aligned} \omega &< \pi/\sqrt{2} \\ T &> 2\sqrt{2} \end{aligned} \quad (4.10)$$

The level of error ξ_{\max} deemed to be acceptable dictates how closely these absolute limits may be approached (§A.2.3).

4.1.3 Image Filtering

Our intention is to apply Gabor kernels to image analysis. Images are filtered by linear convolution forming complex output R :

$$\begin{aligned} R_{\text{continuous}}(x, y) &= \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} I(x + \zeta, y + \eta) q(\zeta, \eta) d\zeta d\eta \\ R_{\text{discrete}}(x, y) &= I * q \end{aligned} \quad (4.11)$$

where $*$ represents discrete linear convolution, I the image and q the Gabor kernel. Discrete convolution is used for real applications. Truncation of the kernel and quantisation of coefficients both form additional sources of error but may be controlled and will be assumed negligible. The extent of the kernel required to avoid severe truncation error may be selected with reference to the space constant of the Gaussian envelope.

Our convolution equations (4.11) do not specify what adjustments should be made at image borders, where pixels have a reduced neighbourhood, and it is often helpful to approximate the image boundary as periodic in order to reduce spurious edge effects. Filter output is unreliable when the kernel extends over textures of more than one type, however, and should ideally be disregarded in these circumstances (Hsiao & Sawchuk, 1989).

Gabor filter output is composed of real and imaginary parts, contributed by the two phases of the kernel: $R = R_c + jR_s$. These may be "demodulated" to give amplitude and phase envelopes:

$$\begin{aligned} A(x, y) &= \sqrt{R_s^2 + R_c^2} \\ \phi(x, y) &= \arctan(R_s/R_c). \end{aligned} \quad (4.12)$$

It should be borne in mind, however, that q_s and q_c are not true quadrature filters, and hence that “amplitude” and “phase” are approximate descriptions. We shall make use of the texture energy \mathcal{R} :

$$\mathcal{R}(x, y) = R_s^2 + R_c^2. \quad (4.13)$$

Although derived from linear convolution, it is important to note that A , \mathcal{R} and ϕ are not themselves linear, and hence linear superposition does not apply. Image filtering must be non-linear in order to extract information because linear transformations are invertible (Caelli, 1985).

Efficient Convolution

Conventional sequential computers are not well-suited to performing linear convolution. Parallel machines achieve higher bandwidth and lower unit cost, and translational symmetry is exploited conveniently by a data-parallel architecture (§A.1.1). Special-purpose hardware may be more appropriate than general-purpose computers for a well-constrained task, further improving the price-performance ratio. It is reasonable to assume that a suitable parallel machine would be able to filter images in real time, and our applications of Gabor filtering have assumed the convolution overhead to be negligible.

At a developmental stage, however, special-purpose hardware is unavailable, and convolution must be performed on a general-purpose computer. Filtering an image of dimensions $n \times n$ pixels with a kernel of size $m \times m$ (usually $m < n$) requires $\mathcal{O}(n^2 m^2)$ operations. Convolution must be repeated for each set of filter parameters, imposing a heavy computational burden. Separability of Gabor kernels along x' and y' -axes may be exploited to reduce this (Heeger, 1987):

$$q(x, y) = \underbrace{\frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{x'^2}{2\sigma^2}\right) \exp(j\omega x')}_{\text{1-D}} \cdot \underbrace{\frac{\lambda}{\sigma\sqrt{2\pi}} \exp\left(-\frac{y'^2 \lambda^2}{2\sigma^2}\right)}_{\text{1-D}}.$$

We may replace the full 2-D convolution by two 1-D convolutions, reducing the complexity to $\mathcal{O}(n^2 m)$.

A more promising approach is to replace convolution in the spatial domain by multiplication in the frequency domain:

$$\mathcal{F}(I * q) = \mathcal{F}(I) \cdot \mathcal{F}(q) \quad (4.14)$$

where $\mathcal{F}(\cdot)$ denotes Fourier transform. Using the FFT algorithm, convolution may now be achieved in $\mathcal{O}(n^2 \log n)$ operations in the best case — this is still large, but is a considerable improvement on naïve convolution. The filter transform $Q = \mathcal{F}(q)$ may be computed directly from (4.7). Use of discrete Fourier transforms implies toroidal boundary conditions.

4.1.4 Gabor Image Representations

Once the existence of elementary functions conveying information with maximum efficiency had been established, it was natural for Gabor (1946) to question whether they defined a basis on which an arbitrary signal could be represented, in order to achieve his goal of compressed transmission and reconstruction of audio signals. In the context of image processing, the objective is to form a compact representation to facilitate subsequent analysis.

An arbitrary image $I(x, y)$ may be represented exactly by the sum:

$$I(x, y) = \sum_{r,s,m,n \in \mathbb{N}} \iota_{m,n}(r, s) q(x - rT_{s,x}, y - sT_{s,y}, \omega_{m,n}, \theta_{m,n}) \quad (4.15)$$

for the parameterisation $q(x, y, \omega, \theta)$, where

$$\begin{pmatrix} \cos \theta_{m,n} & \sin \theta_{m,n} \\ -\sin \theta_{m,n} & \cos \theta_{m,n} \end{pmatrix} \begin{pmatrix} \omega_{m,n} \\ 0 \end{pmatrix} = \begin{pmatrix} \pi\omega_{s,x} \\ \pi\omega_{s,y} \end{pmatrix}$$

provided

$$\begin{aligned} \omega_{s,x}T_{s,x} &= 2\pi \\ \omega_{s,y}T_{s,y} &= 2\pi. \end{aligned} \quad (4.16)$$

Note that in this expansion, the shape of the Gaussian envelope, determined by σ and λ , does not change. The image I is represented by clusters of coefficients $\{\iota\}$ at grid points with separation vector $(T_{s,x}, T_{s,y})^T$; when sampling is isotropic, we may write $T_{s,x} = T_{s,y} = T_s$. Unfortunately, Gabor functions are not orthogonal, and exact determination of $\{\iota\}$ is somewhat involved (Porat & Zeevi, 1988), but they may be determined to a good approximation by convolution when the frequency overlap between adjacent filters is not large (Bovik *et al.*, 1990; Daugman, 1985):

$$\iota_{m,n}(r, s) \simeq I(x, y) * q(x - rT_{s,x}, y - sT_{s,y}, \omega_{m,n}, \theta_{m,n}). \quad (4.17)$$

In practice, manipulation of infinite sets of coefficients is clearly infeasible, and the series is truncated after a few terms to form an approximate representation. When the image is band-limited, the series is finite in any case.

Since all Gabor functions achieve optimal joint uncertainty, there is no need to sample isotropically with a single kernel: Porat and Zeevi (1988) describe a generalised Gabor representation in which ω_s and T_s vary as a function of position (or eccentricity). Their product remains constant, as (4.16), and so a loss-less representation is still possible. In any Gabor representation, the Gaussian envelope retains a constant shape while the sinusoid frequency varies.

Links with the Wavelet Transform

As discussed in §3.5, multi-resolution representations have many advantages for visual information processing because they allow regions of interest to be located approximately at coarse resolution with little computational effort, in order to guide a more

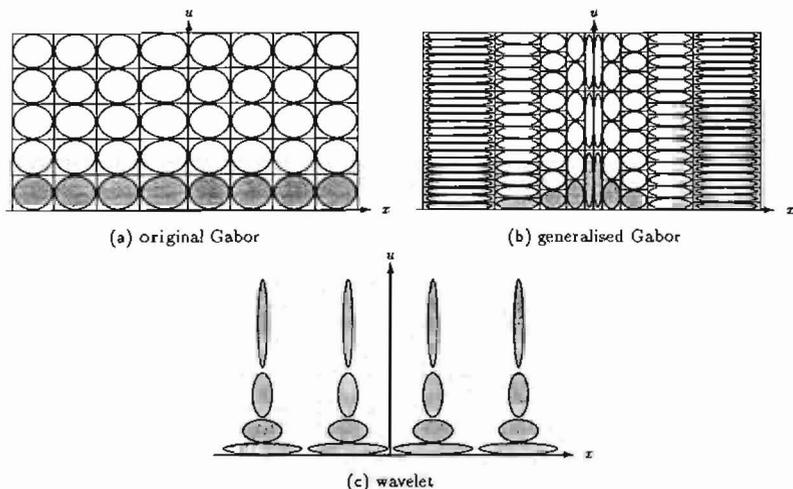


FIGURE 4.3: Information diagram for Gabor and wavelet image representations. Ellipses represent the region sampled by each Gabor kernel, and all have the same area, and x and u represent one-dimensional space and spatial-frequency axes. (a) original Gabor transform: isotropic sampling. (b) generalised Gabor transform: the tessellation is complete but anisotropic. (c) wavelet transform: spatial accuracy decreases with scale. Each transform may use Gabor kernels, and is complete if sufficient samples are taken.

detailed selective analysis. The wavelet transform is a popular method for constructing a representation at many scales (Mallat, 1989). A given kernel ψ , satisfying weak conditions, is convolved with the image for several values of a scale parameter s :

$$\psi_s(x, y) = \sqrt{s} \psi(sx, sy). \quad (4.18)$$

Note that all components of the kernel are scaled in proportion, unlike the Gabor transform (4.15) where the shape of the Gaussian envelope does not change. According to the effect of the uncertainty principle (4.2), the effect of this "dilation" of the kernel is to adjust the spectral and spatial resolutions such that their product remains constant. Large s corresponds to a compact kernel, achieving high spatial but poor spectral accuracy; the position is reversed for small s .

Wavelet transforms are interesting in the present discussion because Gabor functions are acceptable wavelet kernels. Three transforms, all based on linear convolution with Gabor kernels, need to be distinguished. The *original Gabor* transform samples the image with fixed resolution in space and spatial frequency (Bovik *et al.*, 1990). The *generalised Gabor* transform varies the balance between spatial and spectral resolution as a function of eccentricity, achieving higher spatial accuracy near the "fovea" (Porat & Zeevi, 1988). A *wavelet* transform varies resolution as a function of spatial frequency, such that the locations of signals composed of high spatial frequencies are determined accurately (Mallat, 1989). These three alternatives are illustrated in Figure 4.3. In all

cases, use of a Gabor kernel ensures that complete representation of an arbitrary signal is possible if the required sampling density (4.16) is observed.

Examination of our parameterisation (4.8) shows that σ and ω vary inversely when frequency and orientation bandwidths remain constant, consistent with a uniform scaling of the Gabor kernel. This shows that a decomposition along the lines we have proposed is a *wavelet* rather than a Gabor transform (Perry & Lowe, 1989). In practice, only a few terms in the series are taken and the representation is not complete.

4.2 Relation to Other Texture Measures

Numerous approaches to texture analysis involve spatial filtering to some degree, as discussed in Chapter 3. The ultimate goal is to deduce *object* rather than image properties. Physical objects are spatially localised, but cannot be segmented or recognised reliably by inspection of first-order properties alone because these statistics may be confounded by variations in lighting and other extrinsic factors (§3.1). Consequently, image analysis demands examination of both spectral and spatial properties.

Some of the simplest texture methods we reviewed (§3.2.4 on page 41) detect image "features" with local isotropic mask profiles (Coggins & Jain, 1985; Zucker *et al.*, 1975). These filter kernels perform a rudimentary spatial-frequency analysis, but are sub-optimal because joint uncertainty (4.2) is not minimised. Gaussian kernels are an exception to this trend (Marr & Hildreth, 1980) but these are simply an isotropic special case of Gabor functions. Methods based on the power spectrum alone (§3.2.2) may also be viewed as a special case of Gabor filtering.

Orientation is an important cue for visual perception (Zucker, 1983), perhaps because arrangements of oriented line segments can be projectively invariant (Lowe, 1987; Marr, 1976). Gabor filters are suitable for extracting oriented features because orientation tuning and bandwidth (4.8) may be adjusted explicitly. Texture energy filters proposed by Laws (Ballard & Brown, 1982) may be viewed as discrete approximations of Gabor kernels (Bovik *et al.*, 1990). Similarly, the filters proposed by Coggins and Jain (1985) are sensitive to both orientation and spatial frequency but lack optimal joint localisation. Good empirical performance was, however, reported for these texture energy measures (§3.2.4), which bodes well for the success of Gabor filtering.

One further theme developed in Chapter 3 was the use of multiple-resolution representations to facilitate efficient analysis. Since Gabor filters have tunable spatial-frequency characteristics, they are also suitable for this purpose, as in the Gabor pyramid (Porat & Zeevi, 1988). It is a pity that Gabor kernels are not orthogonal because this complicates the determination of coefficients: orthogonal wavelet transforms, for example, allow coefficients to be computed exactly by linear convolution (Mallat, 1989). This practical advantage is not great, however, because approximate Gabor coefficients may still be found by convolution (4.17).

In summary, image analysis often requires simultaneous evaluation of spatial and spectral properties in order to draw reliable conclusions about the physical environment. Various approaches have been proposed to perform local analyses (Chapter 3), but Gabor filters have unique advantages for this purpose, achieving optimal joint resolution

in space and spatial frequency, and form a complete basis so that arbitrary signals may be represented by Gabor coefficients. Following extension of Gabor's (1946) theory to two dimensions (Daugman, 1985), both human and machine vision communities have appreciated their potential for image processing.

4.3 Applications of Gabor Filtering: A Review

Texture energy measures (§3.2.4) may be implemented by simple data-independent processing, allowing approximate segmentation of many classes of image to be performed quickly. In this section, we review previous applications of Gabor filters to image segmentation, before describing our own experiments in §4.4.

4.3.1 Models of Pre-Attentive Human Vision

Psychologists often use artificial multi-element displays to examine the performance of the human visual system because their properties may be more closely-controlled than natural textures (§2.2.4). In this section, we review the use of Gabor filtering to mimic human vision with images of this type. We broaden the discussion to include segmentation of natural textures below (§4.3.2).

The attentive-pre-attentive dichotomy described by psychologists (Chapter 2) is consistent with selective scrutiny of image locations under the guidance of a spatially-parallel filtering stage (§2.4.1). It is now recognised that efficient multi-resolution representations may blur the distinction between "parallel" and "serial" modes, but it is common to model pre-attentive human vision largely by data-independent processing. Psychological theories often fail to generate quantitative predictions because of their vague verbal specification (Broadbent, 1987), and unfortunately definitions for the primitive texture elements on which pre-attentive human vision is believed to operate are of this type (§2.2). Recently, it has been claimed that receptive field profiles of simple visual cells are described well by Gabor functions, which has motivated speculation that these may be the primitive currency of human vision (Daugman, 1985). Any proof of this hypothesis is clearly impossible, and it is unlikely that receptive field profiles possess sufficient precision to allow meaningful discrimination between similar kernels, but this notion has drawn attention to the theoretical advantages of Gabor filters.

Pre-attentive vision is concerned with locating image boundaries rather than object recognition. Texton theory states that pre-attentive vision operates by detecting and marking first-order differences in local image properties or "features" (§2.2.1; Julesz, 1981, 1984), and, building on previous attempts (§2.3), Gabor filtering is seen as an attractive model for the extraction of these statistics (Fogel & Sagi, 1989; Turner, 1986). The implied image model is that different textures map into different Gabor features, such that there is smooth variation within a single filtered texture but sharp variation at texture boundaries. Texture edges may then be located in the filtered image by conventional first-order edge detection. Banks of Gabor filters tuned to different combinations of orientation and spatial frequency are commonly employed to cover parameter space evenly because human vision can hardly know in advance which textures will be

present. Raw "modulated" filter output computed by (4-11) contains phase variation not consistent with the assumed model (Clark *et al.*, 1987), and this is usually removed by computing the Gabor energy (4-13) or amplitude (4-12). This non-linearity also permits output from different linear frequency channels to be combined meaningfully, according to Caelli's (1985) framework.

In the absence of any principled strategy to combine filter output, Turner summed Gabor amplitudes from 32 channels (4 orientations \times 4 frequencies \times 2 phases) computed at grid points (Turner, 1986). When applied to artificial displays of the type designed to verify texton theory (§2.2.1) and Julesz's original conjecture (§2.3.2), this model was adequate to form first-order differences between textured regions pre-attentively discriminable by humans. A high degree of variability remained within each region, however, and Turner did not attempt to extract a boundary, suggesting that recursive application of Gabor filtering might be necessary to distinguish textured regions reliably. High variability of the response within a single texture limits the accuracy with which boundaries may be located, but is always present to some degree when features are derived from local image regions. A quantitative model of the effects of element jitter and patterns of shape variation on the homogeneity of filter output had some success at predicting human texture discrimination performance and search asymmetry (Rubenstein & Sagi, 1990).

A second stage of filtering to smooth feature output, so that only large-scale variation remains, was proposed by Caelli (1985), and may be effective when filtered textures differ in their mean values, rather than in their variances as noted by Turner (1986). Smoothing and boundary detection may be combined by Laplacian-of-Gaussian filtering (Marr & Hildreth, 1980), and this approach was adopted by the GGL algorithm, which operated on demodulated Gabor energy from 32 filters (Fogel & Sagi, 1989). The space constant of the Gaussian was large, in the range 6-18 pixels, and caused distortion of the boundary shape but even after such heavy blurring, spurious noise regions remained in the "segmented" image, and were removed by non-maximal suppression.

Beck's "complex channels" model required two stages of Gabor filtering (Beck *et al.*, 1989; Sutter *et al.*, 1989), as suggested by Turner (1986). Instead of demodulating filter output, a local "standard deviation" was computed from the even (cosine) phase alone, for 39 filters (13 frequencies \times 3 orientations). Differences in filter standard deviations for a pair of textures were combined by a weighted sum to estimate texture discriminability but, although some qualitative effects were predicted correctly, systematic discrepancies with human performance were noted, and it was later concluded that the model failed to account adequately for human texture segregation capabilities (Beck *et al.*, 1989).

These simple Gabor models were partially successful in modelling human texture discrimination. In more general terms, they have suggested some deficiencies of this approach, namely the requirement for many different filter kernels, and variability *within* as well as *between* textured regions after filtering. No principled method for the combination of information from different channels to form a single boundary map was proposed, and heuristic smoothing and post-processing was required to form an acceptable border. These difficulties may arise from the use of artificial multi-element

displays, from arbitrary modelling decisions, or from inherent limitations of the Gabor approach.

4.3.2 Segmentation of Natural Textures

The algorithms described above were intended to model human visual processing, and were constrained by notions of “biological plausibility”. No such restrictions apply in this section: each architecture may be optimised to raise efficiency and functionality. One such improvement involves parameter selection — previous models were obliged to sample the image with a large number of filters to ensure a reasonably even coverage of parameter space. Since the images with which performance was gauged contained only two textures, it is very doubtful whether many of these filters contributed usefully to the segmentation process. Having a fairly complete representation after the initial filtering stage was of little advantage, because there was no intuition about how different channels should be combined.

Clark, Bovik and Giesler (1987) proposed a much simpler image model, allowing most of this complexity to be dispensed with. Each texture type present in an image is assumed to possess at least one range of frequencies uniquely, so that the presence of energy in this band acts as an indicator function. Effectively, each texture is modelled by a single spectral component, and may be thought of as a gray-level modulating function. This model is clearly insufficient for recognition of textured regions, but may be suitable for “spontaneous” texture discrimination, and is perhaps no more unreasonable in the context of textured-edge detection than the ideal step-edge often assumed by conventional edge-finding algorithms (Bovik *et al.*, 1990). The frequency bands which characterise each texture may be located by searching for peaks in the global image power spectrum. Bovik, Clark and Giesler (1990) take the rather extreme view that *all* images may be broken into only two texture types, and choose two sets of parameters corresponding to the two largest spectral peaks (unless human intervention suggests otherwise!). Tan’s (1988) algorithm is a little more flexible, and selects parameters corresponding to all “significant” peaks. Note that this procedure is very similar to feature clustering algorithms described above (§3.5.1), and is liable to fail when the dominant energy modes do not discriminate texture types, or when a texture does not have a dominant frequency. These restrictions are reminiscent of trends observed in human vision (§2.2.4; Treisman & Souther, 1985). Splitting algorithms may also fail to detect the presence of smaller regions, and are susceptible to noise (Bhanu & Parvin, 1987).

An image containing strong frequency bands and its power spectrum are shown in Figure 4.4: the distinct spectral peaks seen here facilitate the estimation of appropriate parameters. Note that power spectra display conjugate symmetry³ because of redundancy arising from the loss of phase information. Care must be taken to eliminate DC and low-frequency components, such as intensity gradients, as these may otherwise dominate the spectrum (§A.2.4). Peak location defines filter frequency and orientation; its shape may be used to select bandwidths.

³Throughout this thesis, we have sampled power spectra for display purposes, and hence symmetry may not be completely reflected in our Figures.

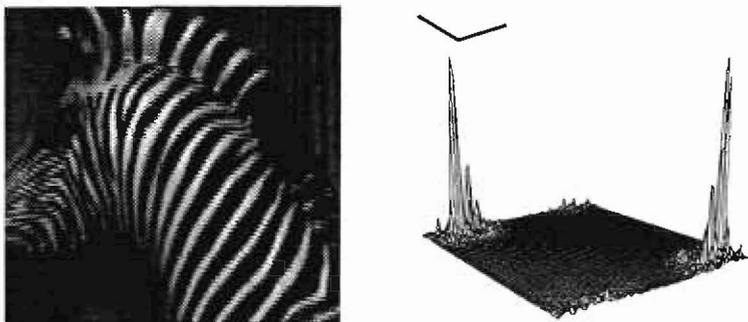


FIGURE 4.4: Zebra image and power spectrum. Suitable Gabor filter parameters may be deduced from the frequency coordinates of peaks in the image power spectrum. Clear peaks emerge in this spectrum (after DC-correction), corresponding to the large diagonal stripes in the zebra's neck. Power spectra display conjugate symmetry.

The difficult problem of channel combination is largely avoided when only a few filters are used. It is possible to perform a per-pixel classification, comparing indicator functions to determine which texture is present at each site (Bovik *et al.*, 1990; Clark *et al.*, 1987):

$$(x, y) \in \mathcal{R}_n, \text{ if } A(u_n, x, y) = \max_i [A(u_i, x, y)] \quad (4.19)$$

where u_i denotes the frequency band of the i -th filter, mapping into region \mathcal{R}_i . We shall refer to this process as “modal filtering”. As with other per-pixel classifications (§3.5.1), the resulting segmentation is noisy but this may be overcome by blurring the texture energy prior to classification in order to eliminate small-scale fluctuations: cleaner boundaries are obtained but there is an accompanying risk of degradation. Artificial and natural Brodatz textures were segmented successfully by this simple method (Bovik *et al.*, 1990; Clark *et al.*, 1987), but it was only demonstrated for periodic textures, and boundaries were sometimes distorted. Generally, modal filtering employs a very simple texture model which may easily break down. Phase discontinuities were detected by an extension to the algorithm, allowing texture boundaries to be located when the dominant frequency mode did not alter (Bovik *et al.*, 1990).

Some of the disadvantages of region processes are overcome by computing edges (§3.5.2), and this approach was adopted by Tan (1988), who computed a feature gradient for each channel. After combination by a weighted sum, thresholding and thinning, accurate connected boundaries were obtained from several synthetic and natural Brodatz textures. These textures appear to have been chosen carefully to be favourable to the algorithm, however, and were mainly periodic with few spectral components, and sufficient area was covered by each texture type to ensure it formed a clear peak in the power spectrum (Tan, 1988).

No real measure of significance is assigned to feature boundaries by the above methods, perhaps because so few channels were chosen, but there is in general a danger that filters capable of discriminating textures accurately will be masked by variability

in other channels. Perry and Lowe (1989) proposed retaining only a compact feature vector, reflecting the maximum response at each site, and normalising variability in each channel by a local standard deviation in order to reduce the impact of “noisy” filters. A relaxation algorithm seeded at the most reliable feature points completed their segmentation algorithm, from which impressive results were reported with collages of natural textures.

Our brief review has suggested that simple algorithms may be found to segment some types of textured images by Gabor filtering but that success is only likely under constrained circumstances, when textures may be treated by very simple models. This type of analysis is valuable to form a preliminary indication of image organisation but is easily confused by textures which do not obey the assumed model. In the next section, we describe some experiments which confirm these impressions.

4.4 Gabor Segmentation Experiments

Despite the attractive theoretical properties of Gabor filters (§4.1), results reported in the literature suggest that simple texture-energy algorithms may be inadequate to segment images reliably, unless they are described accurately by elementary models (§4.3.2). Problems arise because many filters are required to cover parameter space adequately, but few have a beneficial influence on boundary detection, and significant variation of texture energy is observed within as well as between textured regions. In this section, we report our own experiments on Gabor segmentation, and confirm that although Gabor energy filtering is adequate for locating approximate boundaries between simple textures, additional complexity is required to segment more complicated images.

Most experiments were conducted with the simple artificial displays shown in Figure 4.5, designed to test “Segregation” (SEG) and “Pop-out” (POP) ability (§2.1.1). Although highly unrealistic, they allow aspects of segmentation performance to be examined closely. Some real images were also used; all were quantised to 256 gray levels, but no histogram equalisation or normalisation was performed. Filtering was achieved by a mixture of convolution and FFT techniques (page 61), adopting toroidal boundary conditions. The local image mean was adjusted to zero by pre-processing (§A.2.4), in order to ensure that texture energy is not perturbed by the small response to a uniform field generated by all Gabor filters other than pure sine phase.

Notation for specification of filter parameters

Sets of filters covering several orientations are described by the sinusoid period T (pixels), half-height octave spatial-frequency bandwidth $B_{1/2}$, spacing between orientations Θ , and half-height orientation bandwidth $\Delta\theta_{1/2}$. From these may be derived the sinusoid frequency $\omega = 2\pi/T$ (1/pixel units), space constant of Gaussian window σ (pixel units), and window aspect ratio λ , as described in §4.1. Individual filters are specified additionally by their orientation tuning θ , which we restrict to the range $0 \leq \theta < \pi$ because filters $q(\theta)$ and $q(\theta + \pi)$ are equivalent up to a change of sign (4.6 on page 58). Channel outputs were combined from several orientations but a single frequency.

4.4.1 Modal Filtering: A Simple Segmentation Algorithm

Our artificial “Segmentation” and “Pop-out” images (Figure 4.5) are composed of identical line elements which have a thickness of one pixel. In order to respond to these features, we selected a high-frequency filter with a narrow orientation bandwidth: ($T = 3.5$ pixels, $B_{1/2} = 0.4$, $\Delta\theta_{1/2} = \pi/8$; $\sigma = 4.76$ pixels). The sine phase and demodulated amplitude envelope for the “Pop-out” image are shown in Figure 4.6 for the two principal directions along which line elements lie, $\theta = \pi/4$ and $\theta = 3\pi/4$: amplitude demodulation (4.12) has successfully removed the high-frequency phase information, and the location of the singleton element is clearly marked. Filter responses at both these orientations have a high level of residual background variation, but this is eliminated if the line segments are positioned on a regular grid rather than being randomly jittered (data not shown). Responses from filters tuned to orientations $\theta = 0$ and $\theta = \pi/2$ were of low amplitude, suggesting that our filters had been successfully tuned to the diagonal line elements.

Gabor amplitude filtering cannot segment images without further processing, and we computed the mode response (4.19; Bovik *et al.*, 1990) to our artificial images using the filter kernels employed in the previous paragraph at four orientations $\{\theta_i\} = \{0, \pi/4, \pi/2, 3\pi/4\}$, with separation $\Theta = \pi/4$. Taking more filters did not significantly affect the pattern of the results. The effect of *modal filtering* is shown in Figures 4.7 and 4.8: although the singleton element and texture boundary have been detected and successfully characterised, there is some “noise”. This is a feature of many Gabor segmentation programs (Bovik *et al.*, 1990; Fogel & Sagi, 1989; Turner, 1986), as noted above (§4.3). Smoothing with an isotropic Gaussian $g(\zeta)$ eliminates most of the spurious weak responses but the parameter ζ must be chosen carefully: $\zeta = 2$ pixels gives the best result for our “Segregation” image (Figure 4.7) but this would completely obliterate the target element in the “Pop-out” image, for which a smaller value $\zeta = 1$ pixel is optimal (Figure 4.8). A fixed choice of ζ does not seem appropriate, perhaps even within a single image, and some algorithms require this parameter to be adjusted by human intervention (Bovik *et al.*, 1990). Low-pass filtering is a popular technique for textured displays (*e.g.* Bovik *et al.*, 1990; Malik & Perona, 1990; Perry & Lowe, 1989; Sagi & Rubenstein, 1990) because it tends to make the boundaries smoother and to remove isolated weak responses, but is also liable to destroy important detail. Our artificial images highlight some of the deficiencies of modal filtering: it is insensitive and unreliable; may require human intervention to set the smoothing parameter ζ ; and potentially destroys image detail and introduces artifactual boundaries. Simple relaxation algorithms may be used instead of smoothing to regularise the texture boundary, but suffer from similar limitations (Caelli, 1988; Hsiao & Sawchuk, 1989; Reed & Wechsler, 1990).

4.4.2 Appraisal of Feature Contrast Operator

Good discriminant functions have a high level of variation *between* regions rather than necessarily the highest response *within* a region (§3.4.4). Failure to acknowledge this fact was one reason behind the uncertain performance of the power spectrum parameter-selection method (§4.3.2). In order to accentuate relative variations in filter response, we shall introduce the concept of *feature contrast*, formed by weighting amplitude variation.

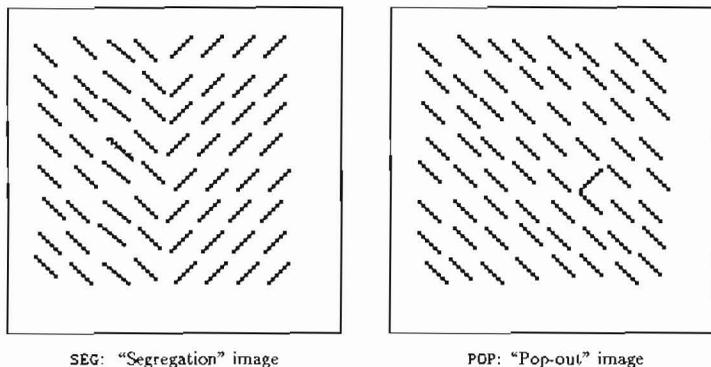


FIGURE 4.5: Artificial “Segregation” and “Pop-out” images. Simple images designed to test texture segmentation algorithms, both 70×70 pixels. Line elements are jittered randomly on a regular grid.

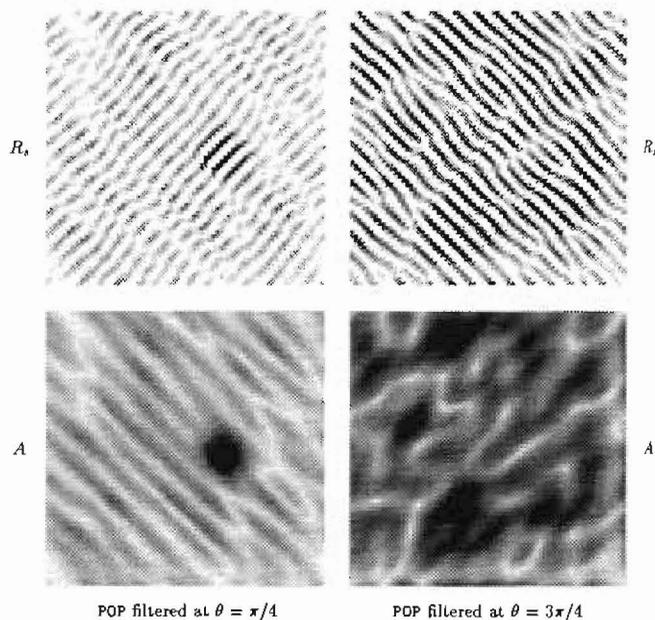


FIGURE 4.6: Raw and demodulated Gabor filter output from “Pop-out” image. Sine phase R_s (positive values only) and demodulated amplitude A obtained from our “Pop-out” image (Figure 4.5) by Gabor filtering at key orientations. Before scaling, the maximum amplitudes were 1:1.88. Demodulation has successfully removed the high-frequency phase information, and the location of the diagonal singleton element is clearly marked. Filter parameters: ($T = 3.5$ pixels, $B_{1/2} = 0.4$, $\Delta\theta_{1/2} = \pi/8$; $\sigma = 4.76$ pixels), using the notation described on page 69.

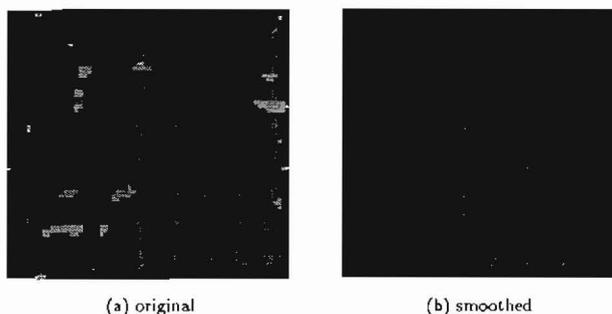


FIGURE 4.7: “Segregation” image after modal filtering. Our “Segregation” image (Figure 4.5) was filtered with Gabor kernels at four orientations and labelled according to the maximum amplitude response (§4.3.2). White represents zero response at all orientations, light grey $\theta = 0$ dominant, up to black for $\theta = 3\pi/4$ dominant. (a) without smoothing; (b) after smoothing with space-constant 2 pixels prior to matching: this gives a “cleaner” boundary. Filter parameters: ($T = 3.5$ pixels, $B_{1/2} = 0.4$, $\Delta\theta_{1/2} = \pi/8$, $\Theta = \pi/4$; $\sigma = 4.76$ pixels), using the notation described on page 69.

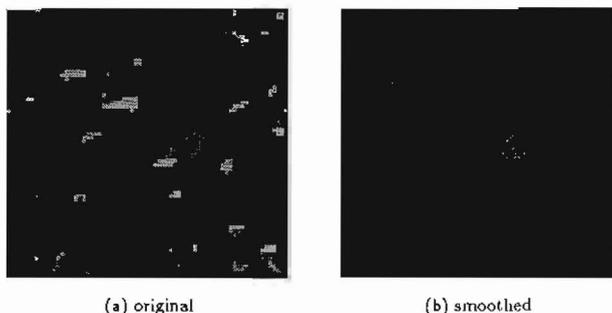


FIGURE 4.8: “Pop-out” image after modal filtering. Our “Pop-out” image (Figure 4.5) was filtered with Gabor kernels at four orientations and labelled according to the maximum amplitude response (§4.3.2). White represents zero response at all orientations, light grey $\theta = 0$ dominant, up to black for $\theta = 3\pi/4$ dominant. (a) without smoothing; (b) after smoothing with space-constant 1 pixel prior to matching: this removes most but not all the “noise”, but smoothing with a larger space-constant also removes the true boundary. Filter parameters: ($T = 3.5$ pixels, $B_{1/2} = 0.4$, $\Delta\theta_{1/2} = \pi/8$, $\Theta = \pi/4$; $\sigma = 4.76$ pixels), using the notation described on page 69.

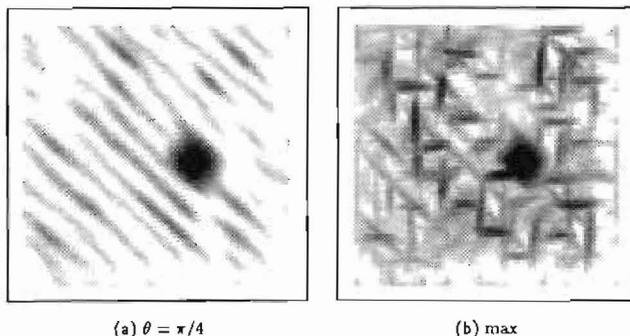


FIGURE 4.9: Effect of contrast operator on filtered “Pop-out” image. Feature contrast was computed for our “Pop-out” image (Figure 4.5) after filtering to extract Gabor amplitude. Untreated output is shown in Figure 4.6. (a) contrast found by (4-20) at orientation $\theta = \pi/4$; (b) overall feature contrast found by (4-21) from four orientations: spurious variation within the background region remains. Filter parameters: ($T = 3.5$ pixels, $B_{1/2} = 0.4$, $\Delta\theta_{1/2} = \pi/8$, $\Theta = \pi/4$; $\sigma = 4.76$ pixels), using the notation described on page 69.

with the intention that places of high contrast gradient map into boundaries. We define contrast $c(\theta, x, y)$ for each channel as:

$$c(\theta, x, y) = \frac{A(\theta, x, y) - \bar{A}(\theta)}{\bar{A}(\theta)} \quad (4-20)$$

finding this form slightly superior to $(A - \bar{A}) / (A + \bar{A})$, where \bar{A} denotes a global average response.

As expected, the singleton target in our “Pop-out” image gives high contrast according to (4-20) in the channel tuned to its orientation, $\theta = \pi/4$ (Figure 4.9a), confirming the potential of our contrast operator. Channels are combined by selecting maximum contrast $C(x, y)$ at each position in the display:

$$C(x, y) = \max_i c(\theta_i, x, y) \quad (4-21)$$

where i ranges over all channels. The result of applying this maximum-contrast operator to our “Pop-out” image is shown in Figure 4.9b: the singleton element correctly generates the highest contrast in the filtered image, but unfortunately fluctuations or “noise” in low-amplitude channels are also mapped into high contrast by the contrast operator (4-20), and a “criss-cross” pattern arises from low-amplitude variation in the $\theta = 0$ and $\theta = \pi/2$ channels. The singleton element in the “Pop-out” image is of sufficient prominence to remain apparent in the global contrast map, but peak contrast is lower for our “Segregation” image and the texture boundary is hidden by “noise” (data not shown). It would be possible to add a threshold to the contrast operator (4-20) to eliminate low-amplitude effects, but this is unlikely to prove satisfactory and may be self-defeating if these convey useful information. At the root of the problem lies the uneven filter response to apparently homogeneous textures: any model which does not allow for this is liable to run into difficulty.

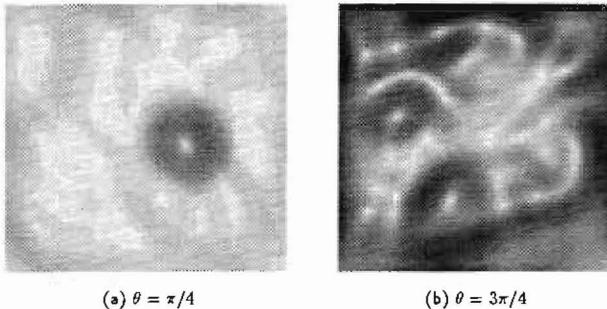


FIGURE 4.10: Effect of gradient operator on filtered “Pop-out” image. Gabor amplitudes of channels tuned to the indicated orientations, filtered by Laplacian-of-Gaussian edge detection with a space-constant of 4 pixels. The original image is shown in Figure 4.5, and untreated output in Figure 4.6. (a) channel tuned to the singleton element, $\theta = \pi/4$; (b) channel tuned to the background elements, $\theta = 3\pi/4$. Spurious background variation in (b) remains, and interferes with attempts to extract a texture boundary. Filter parameters: ($T = 3.5$ pixels, $B_{1/2} = 0.4$, $\Delta\theta_{1/2} = \pi/8$; $\sigma = 4.76$ pixels), using the notation described on page 69.

4.4.3 Appraisal of Feature Gradient Operator

Modal filtering is a region-based process, labelling each image neighbourhood according to its dominant frequency component. Edge-based processes are a suitable alternative (§3.5), and we may locate image sites with high feature gradient because these map naturally into texture boundaries (Tan, 1988). Given the problems with uneven filter response, we have chosen to take the gradient after convolution with a Gaussian kernel:

$$r(\theta, x, y) = \|\nabla g(\rho) * [I(x, y) * q(\theta)]\| \quad (4.22)$$

where the space constant ρ may be adjusted to control the degree of smoothing. Our feature-gradient operator (4.22) is potentially more sensitive to texture boundaries than modal filtering because the requirement that the dominant frequency component must change has been relaxed, and each putative boundary has an associated strength or level of confidence which may be used to eliminate weak responses.

A difficulty with the Gabor energy filter output we obtained previously was the high level of variability within a region of apparently homogeneous texture (Figure 4.6), with fluctuations in amplitude sometimes causing a local drop to a near-zero response. These effects still appear in maps of feature gradient after smoothing ($\rho = 4$ pixels), as shown at key orientations for our “Pop-out” image (Figure 4.10): although high gradient was detected in the channel tuned to the orientation of the singleton element ($\theta = \pi/4$), spurious edges in the orthogonal direction ($\theta = 3\pi/4$) interfere with attempts to extract a valid boundary. Tan (1988) showed that feature gradient is an attractive cue for image segmentation, but the problem of fluctuating filter response must first be understood and either overcome or diminished.

4.4.4 Selection of Filter Parameters and Phase Effects

A consistent problem with the attempts to segment our artificial images reported in this section has been the high level of non-uniformity of filter response to an apparently homogeneous texture, and it becomes particularly acute for our “Pop-out” image (Figure 4.5) where heavy corrective smoothing cannot be employed without destroying detail. Amplitude variation arises because adjacent texture elements may add out of phase, interfering destructively to cause a local drop in response amplitude (Figure 4.11). These effects are very sensitive to line-spacing and the filter period T , perhaps explaining the irregular fluctuations of Gabor amplitude.

Interference between adjacent parts of an image may be reduced by selecting a more local filter (smaller Gaussian envelope). Figure 4.12 shows the response of the filter ($T = 3.5$ pixels, $B_{1/2} = 2$, $\Delta\theta_{1/2} = \pi/4$; $\sigma = 1.1$ pixels) to the same images — previously we used ($B_{1/2} = 0.4$, $\sigma = 4.76$ pixels). The new output contrasts rather sharply with the previous response (Figure 4.6): the general level of variation is much reduced but instead there are local feature “blips” around each texture element. Clearly, the more local filter is responding to individual texture elements rather than to the texture arrangement, but an additional processing step is required in order to join the elements to form composite regions and segment the image. This is contrary to the goals of this approach, which seeks to use spatially-parallel data-independent processing.

When we processed more uniform versions of the artificial “Segregation” and “Pop-out” images, with line segments arranged on a regular grid without jitter, amplitude fluctuations almost disappeared (data not shown) because phase effects were constant across the image. This result confirms our explanation for the source of Gabor amplitude variation within a region of a single texture type. It was fortuitous that both natural and synthetic textures segmented by the algorithms described in the literature and above (§4.3.2) did not display irregular local variation of this type unless heavy smoothing could be tolerated to correct a “noisy” boundary (Bovik *et al.*, 1990; Tan, 1988). Although the feature contrast or feature gradient techniques described above (§§4.4.2–4.4.3) might permit texture borders to be obtained a little more sensitively, it is the limitations of the basic texture model adopted by all these approaches which ultimately constrain performance.

4.4.5 Segmentation of Real Textures

Simple artificial textures specifically designed to expose limitations of segmentation accuracy were employed in the experiments described above (§§4.4.1–4.4.4). We found that these images were not segmented well by previous Gabor texture-energy algorithms because local variations in phase caused fluctuations in Gabor amplitude sufficient to interfere with attempts to extract valid boundaries. Although useful to probe the restrictions of this approach, our “Segregation” and “Pop-out” images (Figure 4.5) are very unrealistic, and it is doubtful whether such patterns of variability occur frequently in practice. In this section, we apply the modal filtering segmentation algorithm (Bovik *et al.*, 1990; Clark *et al.*, 1987) to some real textures, which may obey its simple image model more closely.

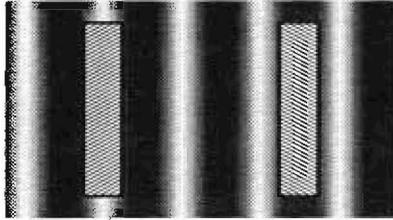


FIGURE 4.11: **Phase effects with Gabor filters.** The Gabor filter illustrated by the background is tuned to the correct frequency and orientation but records low amplitude because the two line segments are out of phase and interfere destructively.

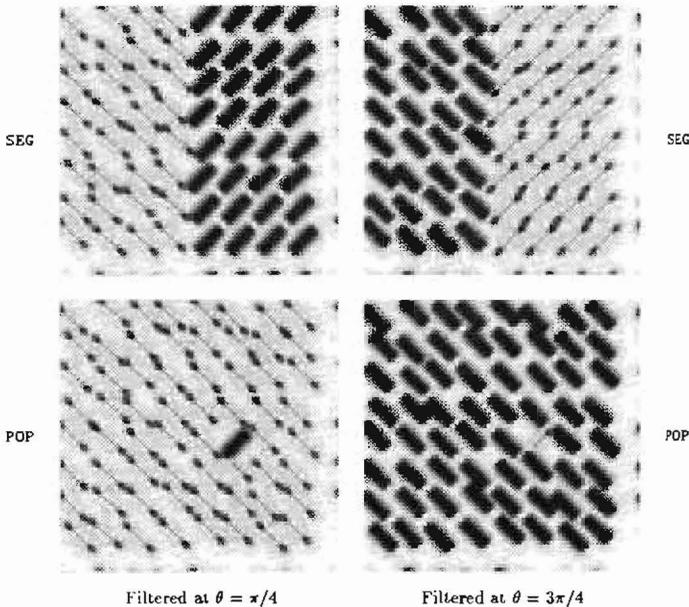


FIGURE 4.12: **Artificial images after filtering with a local kernel.** Demodulated Gabor amplitude from “Segregation” and “Pop-out” images (Figure 4.5) obtained with a local kernel (space constant 1.1 pixels). Texture boundaries are evident but post-processing is required to extract them because of the high level of non-uniformity within each display (compare Figure 4.6). Filter parameters: ($T = 3.5$ pixels, $B_{1/2} = 2.0$, $\Delta\theta_{1/2} = \pi/4$; $\sigma = 1.1$ pixels), using the notation described on page 69.

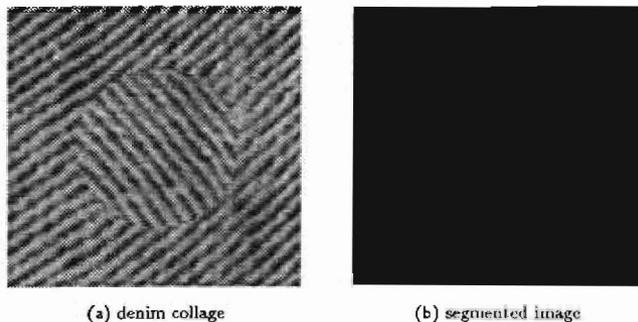


FIGURE 4.13: Denim image after modal filtering. Demodulated Gabor amplitude was obtained from the denim collage by the method described in §4.3.2. (a) denim collage, formed by rotating the central disc by 90° . (b) segmentation produced by modal filtering with Gabor kernels tuned to four orientations and blurring with parameter $\zeta = 2.8$ pixels. Texture contrast is strong enough to overcome this degree of blur. Filter parameters: ($T = 3.5$ pixels, $B_{1/2} = 1.5$, $\Delta\theta_{1/2} = \pi/4$, $\Theta = \pi/4$; $\sigma = 1.4$ pixels), using the notation described on page 69.

The modal filtering algorithm (§4.3.2) assumes that each texture contains a unique dominant frequency component, so that Gabor channels tuned to these critical wavebands may act as indicator functions and segment the image (Bovik *et al.*, 1990; Tan, 1988). Subjectively, the denim texture shown in Figure 4.13a appears similar to a sinusoid, confirmed by the presence of a single sharp peak in the power spectrum of the original texture (data not shown), and hence should be in close correspondence with this texture model. Our collage was formed by rotating the central disc clockwise by 90° so that the diagonal patterns of the material are orthogonal. Both the smooth boundary shape and good separation of texture frequencies are favourable to the segmentation algorithm. We filtered the denim collage with four high-frequency Gabor kernels, and blurred the demodulated channel amplitude prior to extracting boundaries. As shown in Figure 4.13b, the regions extracted have the correct form and general shape, and the estimated boundary lies close to the true contour, although the parameter of the blur kernel was adjusted carefully to get this result ($\zeta = 2.8$ pixels).

Filtering at a single frequency and fixed orientations may be appropriate for images composed of a single base texture (such as Figure 4.13) but is less suitable for discriminating arbitrary textures. The power spectrum parameter-selection method (Bovik *et al.*, 1990; Tan, 1988) is more flexible and often suggests better discriminants, and we adopted this procedure in order to segment the texture montages shown in Figures 4.14 and 4.15, formed by superimposing a patch of one texture over another so that the true boundary was a circle or a diamond (see also Figure 1.3 on page 5). Apart from the denim image used previously, textures were obtained from our Brodatz image set (Figure 1.6 on page 11). These textures often have peak energies at similar frequencies, with periods in the range 5–10 pixels, and are discriminated most effectively if they differ in orientation. All the montages shown in Figure 4.14 are of this type, and were segmented

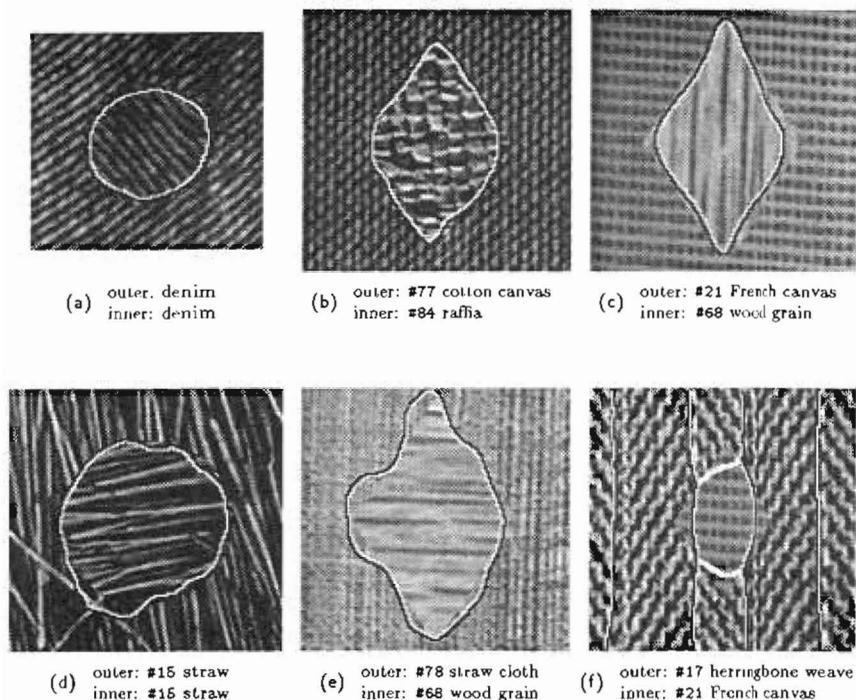


FIGURE 4.14: **Successful Gabor energy segmentations.** Each montage image was filtered with tuned Gabor kernels, two for (a–e) and three for (f), with parameters suggested by the power spectrum method (§4.3.2), and blurred with a space constant of five pixels. The superimposed border marks locations where the dominant response changes, and should form a circle or a diamond. (a) denim image, 108×106 pixels; (b–f) Brodatz images (figure 1.6 on page 11), 128×128 pixels. The segmentation takes the correct form in each case, although boundary shape is sometimes distorted (see also Figure 1.3 on page 5). Filter parameters for (a–e) are shown below, using the notation described on page 69.

image	outer				inner			
	T	θ	$B_{1/2}$	$\Delta\theta_{1/2}$	T	θ	$B_{1/2}$	$\Delta\theta_{1/2}$
(a)	5.2	51°	1.0	25	5.2	141°	1.0	25
(b)	4.5	0°	1.0	25	8.5	86°	1.7	40
(c)	7.1	93°	1.0	25	10.5	5°	1.5	25
(d)	6.0	169°	1.4	35	6.0	79°	1.4	35
(e)	4.4	2°	1.0	25	10.6	95°	0.4	10

well. First-order properties were not normalised but these are not exploited by our algorithm. One Gabor kernel was tuned to each texture patch except #17 (herringbone) where two orientations were used (Figure 4.14f): this is one case where the dominant frequency model breaks down because *two* frequency bands were required to represent this texture. For all these montages, Gabor amplitude was smoothed with a Gaussian of space-constant 5 pixels in order to eliminate small “noise” regions from the labelling procedure (4.19 on page 68). As can be seen from Figure 4.14, this technique was successful at forming a single connected boundary but has tended to distort its shape. Border accuracy could have been improved by adjusting the smoothing parameter for each image.

Some images for which the segmentation algorithm is less successful are shown in Figure 4.15. Failure was caused by: (a) similar frequencies and orientations being chosen to represent each texture, so that they were poorly discriminated; (b,c) textures not possessing a dominant frequency band; (d) texture inhomogeneity. Although some improvement might follow from combining amplitudes from a larger number of channels, or examining feature gradients (Tan, 1988), these failures arise from fundamental weaknesses of the model and cannot be overcome entirely. In fairness, some of the textures in Figure 4.15 are also difficult for humans to distinguish quickly.

4.4.6 Conclusions: Limitations of Gabor Energy Filtering

Gabor filtering can only be the first step in an image segmentation algorithm because Gabor representations are complete and hence re-express image data rather than processing it. Simple image segmentation algorithms suggested in the literature (§4.3) were studied in this section, based on an idealised texture model and demanding only a small number of tuned Gabor channels (Bovik *et al.*, 1990; Tan, 1988). Performance was measured for synthetic and real textures, but only partial success was achieved with both image types. Jitter in our artificial multi-element displays (Figure 4.5 on page 71) led to local variations in filter phase, causing large fluctuations in Gabor amplitude which obstructed attempts to locate texture boundaries. The simplistic texture model predicts that each channel has a constant response within each textured region. Some of the fluctuations noted above may be reduced by smoothing to agree with this assumption more closely, but this process also tends to distort boundary shape, and may destroy useful information.

When suitable filter parameters were chosen, high segmentation accuracy was achieved with montages of natural textures provided these were in close agreement with the assumed texture model. All these texture types were spontaneously discriminable by humans, and resemble those employed previously to demonstrate the potential of this segmentation method (Bovik *et al.*, 1990; Tan, 1988). Poor segmentation accuracy was obtained when the texture model broke down, which could happen for several reasons (Figure 4.15). Failure to segment these images accurately does not arise from a lack of available information because the algorithms developed in Chapter 6 distinguish all these textures reliably, and is not a consequence of insensitive filters since Gabor kernels achieve optimal resolution (§4.1). Poor performance is caused by an incomplete segmentation algorithm arising from an over-simplified texture model.

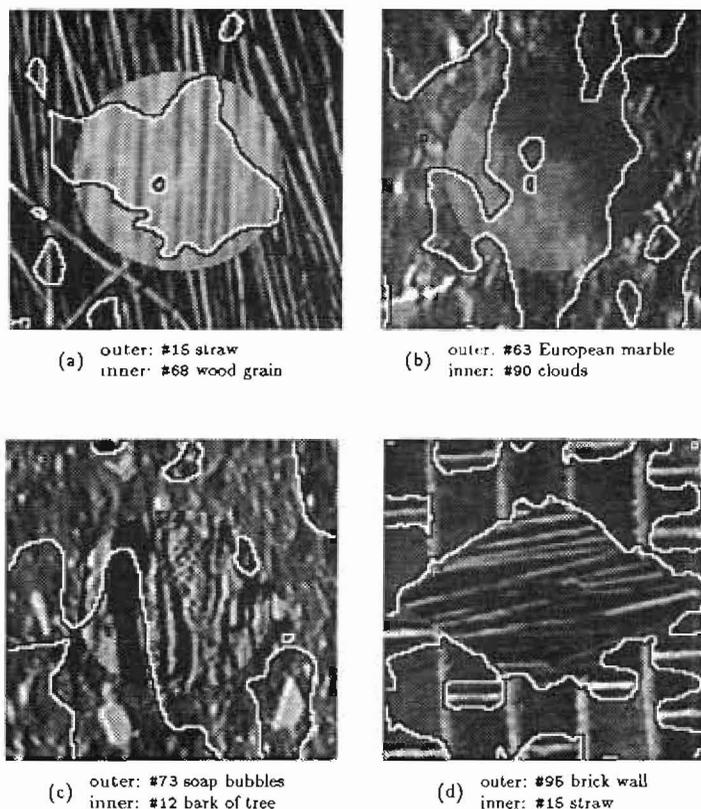


FIGURE 4.15: **Unsuccessful Gabor energy segmentations.** Each Brodatz montage, derived from Figure 1.6, was filtered with two tuned Gabor kernels with parameters suggested by the power-spectrum method (§4.3.2), and blurred. The superimposed border marks locations where the dominant response changes, and should form a circle or a diamond. (a-c) blur space constant five pixels. (d) blur space constant one pixel. The algorithm has become confused because the textures lack a unique frequency band. Filter parameters are shown below, using the notation described on page 69.

image	outer				inner			
	T	θ	$B_{1/2}$	$\Delta\theta_{1/2}$	T	θ	$B_{1/2}$	$\Delta\theta_{1/2}$
(a)	10.8	169°	1.0	30	10.5	5°	1.0	25
(b)	13.4	32°	2.0	40	15.0	144°	2.0	40
(c)	15.7	150°	2.0	40	5.8	5°	1.0	30
(d)	25.1	169°	2.0	45	6.0	79°	2.0	45

4.5 Gabor Signatures: Parameterised Image Models

Our experiments with both artificial and Brodatz textures reported in §4.4 confirmed impressions gained from the literature (§4.3): Gabor amplitude filtering provides a fast and efficient method for obtaining approximate boundaries from suitably constrained images, but is unable to segment unconstrained images reliably. Drawing an analogy with Chapter 2, we may think of Gabor filtering as performing a *pre-attentive* analysis: spatially-parallel and efficient, but used only to guide allocation of more sophisticated processes. Perhaps it is not surprising that simple semi-automatic algorithms using linear filtering are unable to distinguish reliably the complex spatial arrangements we call “texture”. Given their superior theoretical properties (§4.1), we believe that Gabor filters perform at least as well in the role of linear feature-extractor as any similar method reviewed in Chapter 3.

Limitations of Gabor filtering algorithms arise from over-simplified texture models: textures are commonly assumed to form regions of constant or slowly-varying Gabor amplitude, but considerable variation was observed within filtered textures in practice. Some authors suggested recursive application of Gabor filters in an attempt to make texture boundaries more distinct (Turner, 1986; Beck *et al.*, 1989), but this is a rather haphazard procedure. We adopt an alternative approach which seeks to exploit rather than suppress spatial variation in Gabor amplitude, describing spatial arrangements of Gabor features by random fields: our hierarchical model allows us to exploit the speed and flexibility of Gabor filtering combined with the rigorous approach to stochastic spatial dependence offered by random field models. We anticipate that our new model will be more sensitive and reliable than Gabor filtering alone, but more computationally attractive than many structural approaches to texture analysis (§3.3).

4.5.1 Objectives: Concise Feature Representations

Our objectives for the remainder of this Chapter are to convert voluminous Gabor amplitude obtained by image filtering and demodulation into a compact form which may be described conveniently and efficiently by random fields, without severe loss of information. We intend to hide details of Gabor filtering from the random field models, providing a degree of abstraction, by pooling information from several Gabor channels and re-expressing it in terms of a few local images “features”. Unlike some previous approaches described above (§4.3), we intend to derive this transformation in a principled manner. A degree of data compression is required, hence it is inappropriate to use a complete Gabor representation (§4.1.4).

Our approach is to *parameterise* Gabor filter output according to simple local image primitives, described in §4.5.2. At each image site, feature parameters which best describe the observed Gabor signature⁴ are estimated, forming a compact representation of activity in local channels. Several approaches to the estimation procedure are

⁴The signature of a texture primitive is its response spectrum across Gabor filter channels.

described below (§4.6; §A.3). A residual error is computed at each site, enabling a level of confidence to be attached to each feature vector.

Several filter measurements are required in order to estimate feature parameters reliably. Families of filter kernels may be developed in two ways: varying either orientation or spatial frequency (§4.1.2). Orientation is thought to be a key feature in low-level vision (Zucker, 1983), and in order to ensure it is represented accurately, we filter the image with kernels tuned to several orientations $\{\theta_i\}$, where $\theta_i - \theta_{i-1} = \Theta$. Filtering at many spatial scales is computationally attractive, but is not currently implemented — some advantages of an extension to multiple scales are discussed in Chapter 8. It is also possible to take advantage of spatial coherence by pooling filter output from a local neighbourhood, in order to form more reliable feature vectors (Kass & Witkin, 1985), but this approach is inappropriate in the present context because we wish to preserve local spatial variation for analysis by the random field model.

4.5.2 Parameterised Image Primitives

Previous approaches have combined output from several channels in an expedient manner, without any particular image model (§4.3). We take a more principled view: measured channel outputs form an observed Gabor “signature”, which is matched against that of parameterised image primitives. The parameter set which brings these into closest correspondence is used as a feature vector to represent the response at that site. This approach has the advantage that it operates independently of the number or nature of channels used, since the features obtained take a similar form in each case. Care must be taken in designing image primitives: they should have sufficient variation to capture important local structure but must be described by only a small number of parameters, in order that these may be estimated efficiently and reliably.

Currently, we have two primitives, LINE and SINU:

$$f_{\text{LINE}}(x, y) = c \delta(x'' - d) \quad (4.23)$$

$$f_{\text{SINU}}(x, y) = c \sin(\varpi x'' + \psi) \quad (4.24)$$

where the x'' -axis is rotated clockwise by an angle θ_0 from the x -axis (hence $x'' = x \cos \theta_0 - y \sin \theta_0$). Our LINE primitive is a line of vanishing thickness, impulse c , and orientation θ_0 , sited a distance d from the coordinate origin. The SINU primitive is a sinusoid of contrast c , orientation θ_0 , angular frequency ϖ , and phase ψ . Three parameters describe LINE; four describe SINU. Both these image primitives have infinite spatial extent but are suitable for describing local features because of the windowing effect of the Gabor kernel (§4.1.1).

The response to these primitives of a Gabor filter $q(\lambda, \theta, \sigma, \omega)$, parameterised as (4.6) on page 58, is derived in §A.2.5.

LINE:

$$R_s = \frac{c\lambda}{\nu\sigma\sqrt{2\pi}} \exp\left(-\frac{\lambda^2 d^2}{2\sigma^2\nu^2}\right) \sin\left[\frac{\omega d\lambda^2 \cos(\theta_0 - \theta)}{\nu^2}\right] \exp\left[-\frac{\omega^2\sigma^2 \sin^2(\theta_0 - \theta)}{2\nu^2}\right] \quad (4.25)$$

$$R_c = \frac{c\lambda}{\nu\sigma\sqrt{2\pi}} \exp\left(-\frac{\lambda^2 d^2}{2\sigma^2\nu^2}\right) \cos\left[\frac{\omega d\lambda^2 \cos(\theta_0 - \theta)}{\nu^2}\right] \exp\left[-\frac{\omega^2\sigma^2 \sin^2(\theta_0 - \theta)}{2\nu^2}\right] \quad (4.26)$$

$$\mathcal{R} = \frac{c^2\lambda^2}{2\pi\nu^2\sigma^2} \exp\left(-\frac{\lambda^2 d^2}{\sigma^2\nu^2}\right) \exp\left[-\frac{\omega^2\sigma^2 \sin^2(\theta_0 - \theta)}{\nu^2}\right] \quad (4.27)$$

$$\tan\phi = \tan\left[\frac{\omega d\lambda^2 \cos(\theta_0 - \theta)}{\nu^2}\right] \quad (4.28)$$

SINU:

$$R_s = c \cos\psi \exp\left(-\frac{\sigma^2\nu^2\omega^2}{2\lambda^2}\right) \exp(-\sigma^2\omega^2/2) \sinh[\omega\varpi\sigma^2 \cos(\theta_0 - \theta)] \quad (4.29)$$

$$R_c = c \sin\psi \exp\left(-\frac{\sigma^2\nu^2\omega^2}{2\lambda^2}\right) \exp(-\sigma^2\omega^2/2) \cosh[\omega\varpi\sigma^2 \cos(\theta_0 - \theta)] \quad (4.30)$$

$$\mathcal{R} = c^2 \exp\left(-\frac{\sigma^2\nu^2\omega^2}{\lambda^2}\right) \exp(-\sigma^2\omega^2) \left\{ \sin^2\psi + \sinh^2[\omega\varpi\sigma^2 \cos(\theta_0 - \theta)] \right\} \quad (4.31)$$

$$\tan\phi = \cot\psi \tanh[\omega\varpi\sigma^2 \cos(\theta_0 - \theta)] \quad (4.32)$$

Where:

$$\nu^2 = \sin^2(\theta_0 - \theta) + \lambda^2 \cos^2(\theta_0 - \theta), \quad (4.33)$$

and $R = R_c + jR_s$ is the complex channel output (4.11), \mathcal{R} the demodulated Gabor energy (4.13), and ϕ the phase (4.12). Any of these quantities may be used during feature estimation but they are all ultimately derived from the channel response R .

Signature of LINE primitive

Gabor signatures of the LINE primitive are parameterised by: line offset d , orientation θ_0 , and contrast c (4.25–4.28). Typical LINE response curves are shown in Figures 4.16 and 4.17, and response surfaces in Figure 4.18. Gabor output R_{LINE} is symmetric for orientation tuning either side of the true value, $\theta = \theta_0$. If the sign of the line offset d is changed, sine phase R_s is negated but cosine phase R_c is unaffected: for $d \rightarrow -d$, $R \rightarrow R^*$. As orientation error $\vartheta = \theta_0 - \theta$ varies, $\nu^2(\theta)$ (4.33) changes only a little for a nearly circular Gabor envelope, aspect ratio $\lambda \approx 1$, and the major sources of variation of response R with ϑ are the terms in $\sin^2\vartheta$ and $\cos\vartheta$ (4.25–4.26). When line offset and Gabor period are comparable ($d > T/4$), R_s and R_c may oscillate mildly with ϑ , but oscillation with line offset d is sharp, as confirmed by Figures 4.17 and 4.18a. As this oscillation is mainly due to a variation in phase, Gabor energy \mathcal{R} is little affected (Figure 4.18b). Further discussion appears in §A.3.1. Channel output R scales linearly with line contrast c .

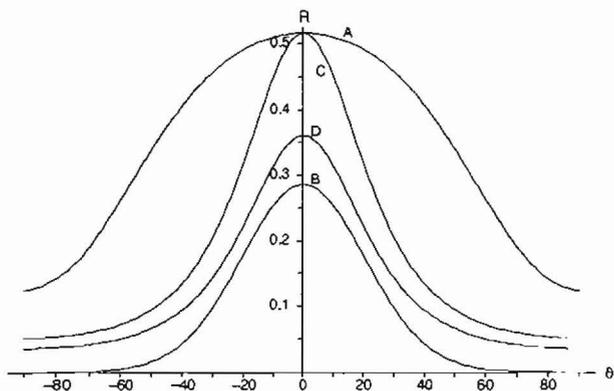


FIGURE 4.16: **Gabor signature of untranslated LINE primitive.** Cosine phase R_c for the response to a LINE primitive (4-23 on page 82) coinciding with filter origin, and relative orientation ϑ (degrees), for typical filter kernels. (a) high orientation-bandwidth, $\Delta\theta_{1/2} = \pi/4$; (b) low frequency-bandwidth, $B_{1/2} = 1$; (c) high frequency-bandwidth, $B_{1/2} = 2$; (d) low frequency, $T = 5$ pixels. Except where otherwise stated above, filter parameters are: ($T = 3.5$ pixels, $B_{1/2} = 2$, $\Delta\theta_{1/2} = \pi/8$), using the notation described on page 69. The sine phase is zero in all cases.

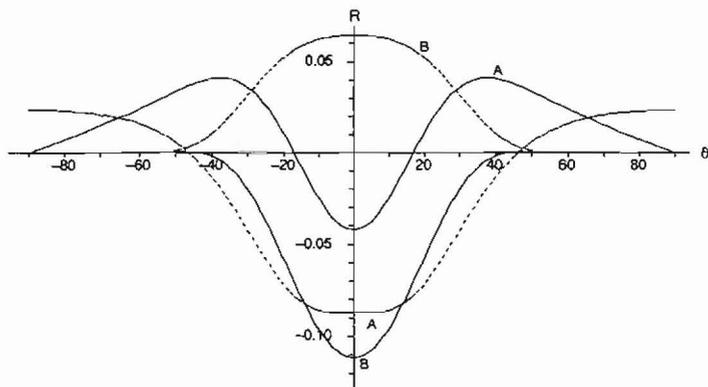


FIGURE 4.17: **Gabor signature of translated LINE primitive.** Cosine phase R_c (dotted) and sine phase R_s (solid) response to a LINE primitive (4-23 on page 82) translated by d pixels from the filter origin, and relative orientation ϑ (degrees), for typical kernels. (a) high frequency-bandwidth, ($T = 3.5$ pixels, $B_{1/2} = 2$; $d = 2$ pixels); (b) narrow frequency-bandwidth, ($T = 3$ pixels, $B_{1/2} = 1/2$; $d = 2.5$ pixels), using the notation described on page 69. Orientation bandwidth was $\pi/8$ for both filters.

Signature of *SINU* primitive

Gabor signatures of the *SINU* primitive are parameterised by: grating frequency ω , orientation θ_0 , contrast c , and phase ψ (4.29–4.32). The response R_{SINU} is symmetric for filters tuned either side of the correct orientation θ_0 . The phase of the grating ψ affects both the phase and amplitude of the channel response, applying a small correction to Gabor energy. Unlike the response to the *LINE* primitive, R_{SINU} does not oscillate: both R_s and R_c take the same sign everywhere, and their amplitude reaches a maximum when the filter is perfectly tuned, to the correct frequency $\omega = \omega$ and orientation $\theta = \theta_0$, with roughly exponential decay when mis-tuned (Figure 4.19). Apart from a scaling factor, R_s and R_c vary in a very similar way, and the Gabor phase $\tan \phi$ (4.32) is nearly constant for $\theta \approx \theta_0$ — when its argument becomes large, $\tanh(\omega\omega\sigma^2 \cos \theta) \approx 1$, and hence $\tan \phi \approx \cot \psi$. Channel output R scales linearly with contrast c .

4.5.3 Least-Squares Feature Extraction

Our intention is to combine measured responses from several Gabor channels, representing them with a small number of feature parameters. These will usually be over-constrained by equations (4.25–4.28) for *LINE* and (4.29–4.32) for *SINU* primitives. Exact inversion of these equations is not feasible in practice, and would also fail to form the “best” estimate from “noisy” measurements; least-squares parameter estimation is a suitable technique to apply in these circumstances (Papoulis, 1990, Chapter 11). Unfortunately, the response equations for both *LINE* and *SINU* primitives are too complicated for us to obtain a closed-form solution by this method, even after simplifying assumptions discussed below, and we currently employ the sub-optimal feature-extraction techniques described in §4.6.

Let us write the measured and model responses from all channels at each image site as complex vectors \mathbf{M} and \mathbf{R} , respectively, one element corresponding to each filter. We may then seek to minimise the error ε between measured and response vectors:⁵

$$\varepsilon = \|\mathbf{M} - \mathbf{R}\|^2. \quad (4.34)$$

For the *LINE* primitive, ε is minimised with respect to the free parameters $R_{\text{LINE}} = R(\theta_0, c, d)$ to generate three equations:

$$\frac{\partial \varepsilon}{\partial \theta_0} = \frac{\partial \varepsilon}{\partial c} = \frac{\partial \varepsilon}{\partial d} = 0. \quad (4.35)$$

In principle, these are solved as simultaneous equations to obtain the matched parameters $\hat{\theta}_0$, \hat{c} and \hat{d} , and to give ε_{min} . Unfortunately, the complicated nature of the variation of R_{LINE} with θ_0 (4.25–4.26) means that a closed-form solution is not possible, and even if the equations are simplified by assuming that the line coincides with the filter kernel,

⁵Where $\|\mathbf{A}\|$ denotes the Euclidean norm: $\|\mathbf{A}\|^2 = \sum_i |A_i|^2$.

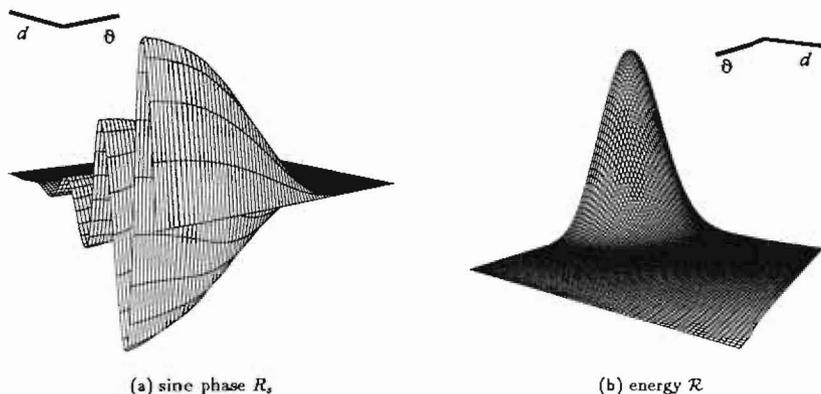


FIGURE 4.18: **Response surfaces for the LINE primitive.** Gabor signature for the LINE primitive (4.23 on page 82) as a function of orientation tuning ϑ and line offset d (pixels), in the ranges $0 \leq \vartheta \leq \pi/2$ and $0 \leq d \leq 15$. (a) sine phase R_s oscillates rapidly with d small; (b) Gabor energy \mathcal{R} varies smoothly. Filter parameters: ($T = 3.5$ pixels, $B_{1/2} = 0.5$, $\Delta\theta_{1/2} = \pi/6$; $\sigma = 3.8$ pixels), using the notation described on page 69.

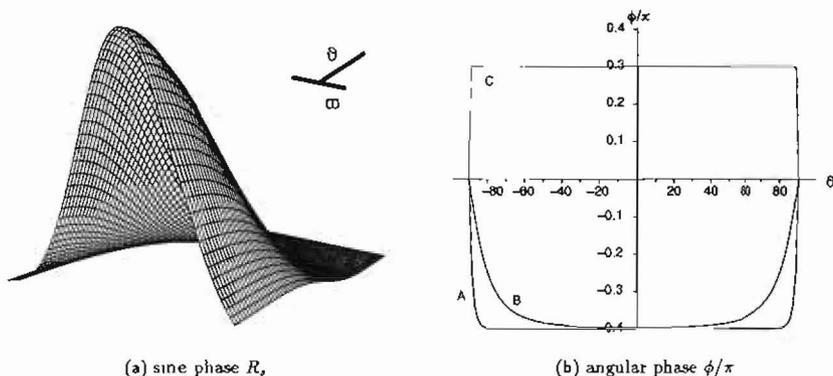


FIGURE 4.19: **Gabor signature of the SINU primitive.** Response to the SINU primitive (4.24 on page 82). (a) response surface for sine phase R_s , as a function of orientation tuning ϑ and sinusoid frequency ω , in the ranges $0 \leq \vartheta \leq \pi/2$ and $0.15\omega \leq \omega \leq 2\omega$, and for grating phase $\psi = 0$. Maximum response is reached when the filter is correctly tuned. (b) Gabor phase ϕ for several combinations of filter and SINU parameters, as a function of orientation tuning ϑ (degrees): (A) $\omega = \omega$, phase $\psi = 0.9\pi$; (B) $B_{1/2} = 2$; $\omega = \omega/2$, $\psi = 0.9\pi$; (C) $B_{1/2} = 0.5$; $\omega = 2\omega$, $\psi = 1.2\pi$. Filter parameters: ($T = 5$ pixels, $B_{1/2} = 1$, $\Delta\theta_{1/2} = \pi/6$; $\sigma = 2.8$ pixels), except where stated otherwise above, using the notation described on page 69.

$d = 0$, we reach a pair of very unwieldy simultaneous equations:

$$\sum_i \frac{M_{c,i}}{\nu_i^2} \exp \left[-\frac{\omega^2 \sigma^2 \sin^2(\hat{\theta}_0 - \theta_i)}{2\nu_i^2} \right] \sin(\hat{\theta}_0 - \theta_i) \cos(\hat{\theta}_0 - \theta_i) [(1 - \lambda^2)\nu_i^2 + \omega^2 \sigma^2 \lambda^2] =$$

$$\frac{\hat{c}\lambda}{\sigma\sqrt{2\pi}} \sum_i \frac{\sin(\hat{\theta}_0 - \theta_i) \cos(\hat{\theta}_0 - \theta_i)}{\nu_i^2} \exp \left[-\frac{\omega^2 \sigma^2 \sin^2(\hat{\theta}_0 - \theta_i)}{\nu_i^2} \right] [(1 - \lambda^2)\nu_i^2 + \omega^2 \sigma^2 \lambda^2]$$

$$\sum_i \frac{M_{c,i}}{\nu_i} \exp \left[-\frac{\omega^2 \sigma^2 \sin^2(\hat{\theta}_0 - \theta_i)}{2\nu_i^2} \right] = \frac{\hat{c}\lambda}{\sigma\sqrt{2\pi}} \sum_i \frac{\exp[-\omega^2 \sigma^2 \sin^2(\hat{\theta}_0 - \theta_i)/\nu_i^2]}{\nu_i^2} \quad (4.36)$$

where

$$\nu_i^2 = \sin^2(\hat{\theta}_0 - \theta_i) + \lambda^2 \cos^2(\hat{\theta}_0 - \theta_i) \quad (4.37)$$

and summation for i extends over all channels. Note that because the sine phase of the model signature R_s is identically zero, estimated features (4.36) depend solely on measured cosine phase $\{M_{c,i}\}$. Exact solution for \hat{c} and $\hat{\theta}_0$ is not practical. The above form assumes that all filter parameters except orientation (that is, ω , σ and λ) are common, and this was true for our experiments, but this restriction need not apply.

Inversion of the responses to the SINU primitive (4.29–4.30) is just as difficult, even if less important parameters are dropped to simplify the problem. Consequently, we rejected simultaneous least-squares estimation of all parameters as infeasible in practice. In §4.6, we describe a sub-optimal estimator based on response moments which generates acceptable features without invoking the full complexity of our parameterised signatures: this is our preferred method. Some alternative approaches are discussed in §A.3.

4.6 Feature Estimation using Response Moments

Least-squares estimation of feature parameters from measured responses of several channels using the parameterised Gabor signatures R_{LINE} (4.25–4.26) and R_{SINU} (4.29–4.30) is not feasible in practice because the equations involved are extremely unwieldy (§4.5.3). In this section, we develop alternative sub-optimal parameter estimators by exploiting response moments. Our Profile and Resultant methods are shown to generate acceptable features efficiently, and both are employed below in conjunction with our hybrid Gabor-Markov texture models (Chapter 6). Some further approaches and our reasons for having rejected them are discussed in §A.3.

4.6.1 Response Moments

From the form of the model response equations for both LINE (4.27) and SINU (4.31) primitives, it is clear that Gabor energy $\mathcal{R}(\theta)$ is symmetric and decreasing about the orientation of the image primitive θ_0 (except possibly for very ill-tuned filters), as confirmed for the LINE primitive by Figures 4.16 and 4.17. We may exploit this simple angular variation by setting the estimated orientation feature $\hat{\theta}_0$ to the angle at which

the angular moment of the observed response energies is a minimum. This may be expressed as a least-squares estimate by minimising the lengths of projections onto the appropriate axis, writing:

$$\varepsilon' = \sum_i \mathcal{M}_i \sin^2(\theta_0 - \theta_i) \quad (4-38)$$

and minimising ε' with respect to the free parameter, θ_0 . This approach is equivalent to treating $\mathcal{R}(\theta)$ as an angular distribution and computing its mean (Kass & Witkin, 1985). We map $\mathcal{R}(\theta)$ around the unit circle, writing:

$$\boldsymbol{\mu} = \int_{-\pi/2}^{\pi/2} \mathcal{R}(\theta) \begin{pmatrix} \sin 2\theta \\ \cos 2\theta \end{pmatrix} d\theta \quad (4-39)$$

where the direction of $\boldsymbol{\mu}$ defines the resultant mean, $\bar{\theta} = \theta_0$, as expected from symmetry of $\mathcal{R}(\theta)$. This is true for both LINE and SINU primitives, and for any other image primitive possessing appropriate symmetry.

In order to estimate θ_0 , we may compute the angular mean (4-39) using the measured Gabor energies $\mathcal{M}(\theta_i)$. Replacing integration by discrete summation, we have

$$\tan 2\hat{\theta}_0 = \frac{\sum_i \mathcal{M}_i \sin 2\theta_i}{\sum_i \mathcal{M}_i \cos 2\theta_i} \quad (4-40)$$

where summation for i extends over all channels. For this to be accurate, it is helpful for the channels to be regularly spaced around the circle, and for their responses to overlap somewhat. This may be arranged by selecting appropriate filter orientations and orientation bandwidths. One further way of expressing the estimator is by the direction of the eigenvectors of the matrix:

$$\sum_i \begin{pmatrix} \cos^2 \theta_i & \sin \theta_i \cos \theta_i \\ \sin \theta_i \cos \theta_i & \sin^2 \theta_i \end{pmatrix} \mathcal{M}_i.$$

Both $\hat{\theta}_0$ and $\hat{\theta}_0 + \pi/2$ are solutions: we choose the value that minimises ε' (4-38). This form for $\hat{\theta}_0$ is much more convenient than attempts to use least-squares estimation (§4.5.3) because it is readily evaluated, and the same equation holds for both LINE and SINU primitives.

Our equation for the estimated orientation feature $\hat{\theta}_0$ (4-40) does not refer directly to the model energy response \mathcal{R} , although the general form of \mathcal{R} was applied indirectly in selecting the moment method. Consequently, our estimated orientation is not necessarily an optimal solution to the overall feature estimation problem (4-35). We have dealt with orientation in this way because it was the feature with the least prospect of being obtained from the observed Gabor signatures by analytic methods.

Note that because $\mathcal{R} = RR^*$, we are effectively computing a second moment of the linear channel response. It is tempting to ask whether higher-order moments could lead to estimators for further parameters, but unfortunately the integration required to compute model responses against which observed values are compared, is intractable, and it is also likely that "noise" would become a serious factor.

4.6.2 Profile Feature Extraction Method

Once the orientation feature $\hat{\theta}_0$ has been obtained by the moment estimator (4.40), we may again consider least-squares estimation to recover the remaining features (§4.5.3). This will, in general, form a sub-optimal feature vector compared with solution of the complete least-squares equation (4.35), but may still be a good approximation. Estimation of *all* remaining feature parameters is still infeasible, but we may simplify the problem by accepting a partial solution. Orientation features for *LINE* and *SINU* primitives are identical because they are both obtained by the moment estimator (4.40), but different treatments are appropriate to extract the remaining features using parameterised Gabor signatures.

Feature Estimation using the *LINE* primitive

Even with the orientation feature $\hat{\theta}_0$ given independently by (4.40), it is still impractical to write down a least-squares solution for both remaining features, line contrast \hat{c} and offset \hat{d} (§A.3.1). Instead, we choose to set $d = 0$ and to disregard the phase of the observed response \mathbf{M} , effectively assuming the image line intersects the filter origin: this is justified because any match away from $d \approx 0$ will have low amplitude and high error. We may now write a least-squares contrast estimator \hat{c}^2 in terms of observed Gabor energies \mathcal{M}_i :

$$\varepsilon = \sum_i \left[\frac{\mathcal{M}_i - \mathcal{R}_i}{\Delta \mathcal{M}_i} \right]^2 \quad (4.41)$$

$$\Delta \mathcal{M} \approx 2\sqrt{\mathcal{M}} \Delta \mathcal{M}$$

where $\Delta \mathcal{M}$ represents the standard deviation of \mathcal{M} , and we have assumed that the measured responses have equal standard deviations: $\Delta \mathcal{M}_c = \Delta \mathcal{M}_s = \Delta \mathcal{M}$. Each term is normalised by its estimated standard deviation in order to give equal weight to each measured quantity. We use Gabor energy rather than the separate phases because this degrades more gracefully when modelling assumptions are inexact, but is insensitive to the *sign* of the line contrast c , which is consequently not determined by this method. Minimisation of ε with respect to c^2 gives:

$$\hat{c}_{\text{LINE}}^2 = \frac{2\pi\sigma^2}{\lambda^2} \frac{\sum_i \exp[-\omega^2\sigma^2 \sin^2(\hat{\theta}_0 - \theta_i)/\nu_i^2] / \nu_i^2}{\sum_i \exp[-2\omega^2\sigma^2 \sin^2(\hat{\theta}_0 - \theta_i)/\nu_i^2] / (\nu_i^4 \mathcal{M}_i)} \quad (4.42)$$

plus the trivial solution $\hat{c} = 0$. When the true line feature does lie some distance d from the centre of the filter, apparent contrast \hat{c}^2 decays roughly as: $\exp(-d^2/\sigma^2)$, under the influence of the Gabor envelope, as is apparent in Figure 4.21 on page 93. Non-maximal suppression or “lateral inhibition” (Malik & Perona, 1990) could perhaps be employed to sharpen the response, but this would probably also remove valid detail and is not currently implemented. The residual least-squares error ε_{min} , obtained by minimising (4.41), represents the level of agreement achieved between measured and model Gabor energy signatures, \mathcal{M} and $\mathcal{R}(\hat{\theta}_0, \hat{c}^2)$, and is used to define an “error” feature $\hat{\varepsilon}_{\text{LINE}}$:

$$\hat{\varepsilon}^2 = \frac{\varepsilon_{\text{min}}}{\|\mathbf{M}\|^2} \quad (4.43)$$

where $0 \leq \hat{\varepsilon} \leq 1$.

Feature Estimation using the SINU primitive

Similar difficulties are experienced with the SINU primitive, where it is necessary to assume the filter and grating frequencies are equal: $\varpi = \omega$; and convenient to set the grating phase to zero: $\psi = 0$. We may then write a least-squares estimator for grating contrast, using the same approach as above (4.41):

$$\hat{c}_{\text{SINU}}^2 = \exp(\omega^2 \sigma^2) \frac{\sum_i \exp(-\omega^2 \sigma^2 \nu_i^2 / \lambda^2) \sinh^2[\omega^2 \sigma^2 \cos(\hat{\theta}_0 - \theta_i)]}{\sum_i \exp(-2\omega^2 \sigma^2 \nu_i^2 / \lambda^2) \sinh^4[\omega^2 \sigma^2 \cos(\hat{\theta}_0 - \theta_i)] / \mathcal{M}_i}. \quad (4.44)$$

When filter and grating frequencies differ, $\varpi \neq \omega$, apparent contrast \hat{c}^2 again falls off approximately exponentially, as: $\exp[-\sigma^2(\omega - \varpi)^2]$, under the influence of the Gabor envelope. We define $\hat{\varepsilon}_{\text{SINU}}$ as above (4.43).

4.6.3 Resultant Feature Extraction Method

Profile feature extraction (§4.6.2) uses our moment estimator (§4.6.1) to obtain the orientation feature only: other features are obtained (in principle, at least) by least-squares Gabor signature matching, which also generates a residual error or level of confidence ε . An alternative approach is to abandon LINE and SINU primitives entirely, and derive *all* features directly from observed channel responses without reference to image primitives: we call this the Resultant feature-extraction method. It has the advantage that features may be computed more simply than above, from (4.42) or (4.44).

The Resultant orientation feature $\hat{\theta}_0$ is computed exactly as above (4.40); only contrast and residual error are treated differently. We define the Resultant “contrast” feature to be the length of the resultant vector formed in (4.40):

$$\hat{c}_{\text{RES}}^2 = \left\| \sum_i \mathcal{M}_i \begin{pmatrix} \sin 2\theta_i \\ \cos 2\theta_i \end{pmatrix} \right\|. \quad (4.45)$$

Unlike previous estimators, which respond mainly to variations in image properties (4.42 and 4.44), this definition is sensitive to changes in filter parameterisation (σ , ω and λ), but this does not matter in practice because we hold these parameters constant, varying only filter orientation θ , and because the spatial arrangement of derived features is of more importance than their first-order properties. Note that we have defined contrast (4.45) relative to the orientation axis θ_0 rather than taking the absolute measure $\|\mathbf{M}\|^2$, to reduce the effects of noise. Unlike the Profile feature-extraction method, Resultant features are not tied to a particular image primitive, and hence there are not separate features for LINE and SINU primitives.

In a similar spirit, we define a Resultant “error” feature $\hat{\varepsilon}_{\text{RES}}$:

$$\hat{\varepsilon}_{\text{RES}} = 1 - \frac{\hat{c}^2}{\|\mathbf{M}\|^2}. \quad (4.46)$$

This measures the extent to which Gabor energy projects in a coherent direction, and varies in a similar manner to the Profile residual error (4.43). Our definition (4.46) is again imperfect because ε approaches zero only in exceptional circumstances, but it does permit an indication of response coherence to be computed quickly.

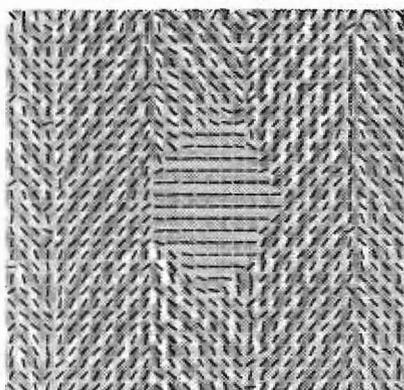
4.6.4 Examples of Features Extracted from Real Textures

Natural textures are too ill-constrained for any theoretical model to be entirely accurate, and empirical verification of performance is often appropriate. We obtained feature sets from a selection of textures in order to assess the success of our signature-matching framework in fulfilling our objectives (§4.5.1), and to compare the performance of different estimators. Each feature vector has three components: orientation, apparent contrast and residual modelling error, as described above. Since these cannot be represented directly, we show features only at fixed grid points, using small needles aligned with the orientation feature; contrast is represented by line length and residual error by intensity (black corresponds to zero error), both on a linear scale. Toroidal boundary conditions were assumed, and hence feature vectors immediately adjacent to image boundaries are unreliable.

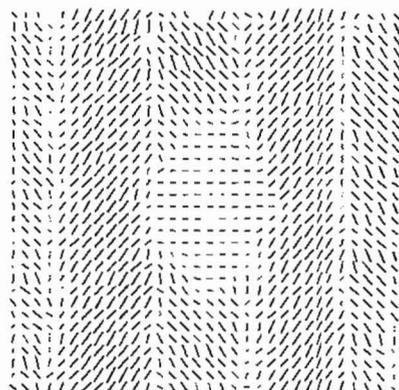
Each image was filtered at several orientations (usually four) but at only a single spatial frequency, and consequently the feature sets do not represent the full hierarchical structure of the texture but only a single layer of it. Effectively, microtexture is captured at only a single scale — extension to filtering at multiple scales is discussed in Chapter 8. Features are displayed at fixed grid points, without non-maximal suppression or hysteresis, and hence do not have as much freedom to follow image contours as typical edge-detector output (*e.g.* Canny, 1983). For our application, it is more appropriate that filters should respond to *regions* than boundaries, providing information on the spatial structure of the texture for analysis by random field models.

Both our Profile and Resultant feature-extraction methods (§§4.6.2–4.6.3) rely on response moments to derive the orientation feature (§4.6.1), and hence the ability of this estimator (4.40 on page 88) to respond sensitively to local image structure is of paramount importance. Using needles of unit length and intensity, we show the orientation feature map extracted from the herringbone-canvas montage (Figure 4.14*f* on page 78) in isolation in Figure 4.20*a*, having deduced suitable filter parameters by the power-spectrum method (§4.3.2). Comparison with the original texture confirms that small variations in the orientation feature accurately reflect local texture structure, following small irregularities in the weave in this case. Successful segmentation of this montage (Figure 4.14*f*) confirmed that Gabor features are suitable for determining gross texture structure; this new result suggests that they are also sensitive to fine variation. Similar conclusions may be drawn from other examples described below.

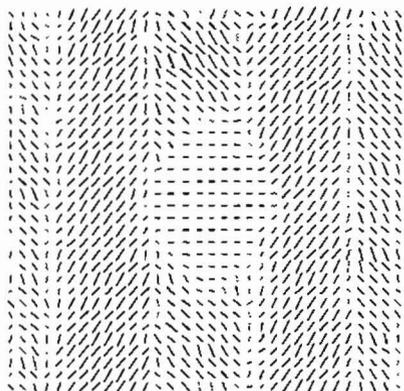
Contrast and error features extracted by Profile (LINE and SINU) and Resultant estimators are shown in Figure 4.20*b–d*; in each case, the orientation is the same as that represented in (*a*). These features again respond to local texture structure, appearing fainter (low confidence) when evidence for the selected orientation is inconclusive, notably near texture boundaries. Small differences between the three sets of features



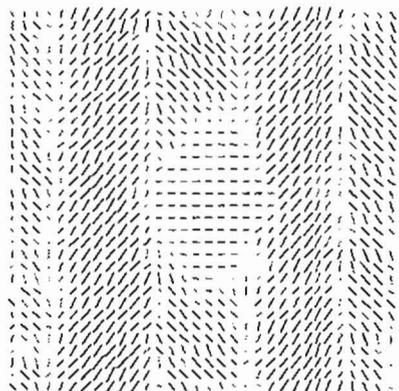
(a) estimated orientation



(b) LINE features



(c) SINU features



(d) Resultant features

FIGURE 4.20: Gabor features extracted from herringbone-canvas montage. Using parameters suggested by the power-spectrum method (§4.3.2), Gabor feature vectors were estimated for the montage shown in Figure 4.14f on page 78, and sampled every four pixels. Needle intensity represents “error”, and length is proportional to “contrast”. (a) orientation feature estimated by our moment estimator (common to each feature set); (b) Profile LINE features; (c) Profile SINU features; (d) Resultant features. Only small differences may be distinguished. Filter parameters: ($T = 7.2$ pixels, $B_{1/2} = 0.7$, $\Delta\theta_{1/2} = \pi/6$, $\theta = \pi/4$; $\sigma = 5.7$ pixels), using the notation described on page 69.

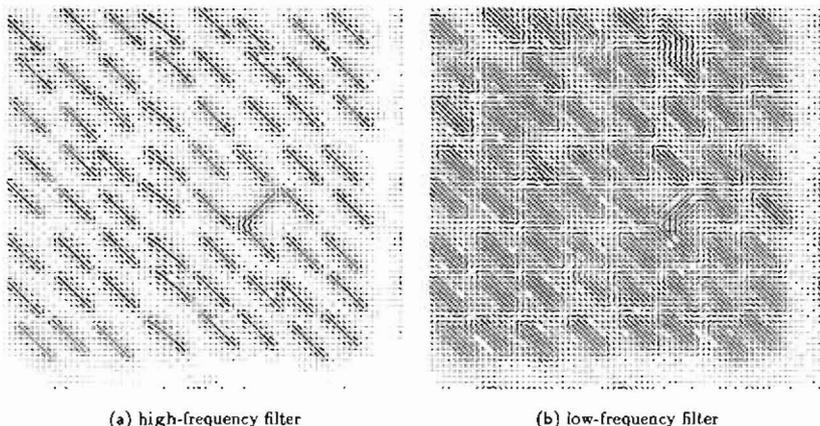


FIGURE 4.21: Gabor features extracted from “Pop-out” image. Feature vectors were estimated from the “Pop-out” image (Figure 4.5 on page 71) using the Profile LINE method. Needle intensity represents “error”, and length is proportional to “contrast”. (a) high-frequency filter: ($T = 3.5$ pixels; $\sigma = 1.1$ pixels); (b) low-frequency filter: ($T = 7$ pixels; $\sigma = 2.2$ pixels). Feature blurring becomes more pronounced at lower spatial frequencies but may be partially offset by sampling. Other filter parameters: ($B_{1/2} = 2$, $\Delta\theta_{1/2} = \pi/4$, $\Theta = \pi/4$), using the notation described on page 69.

may be distinguished, particularly in the central (canvas) region, but they are remarkably similar given that different modelling assumptions apply in each case. Only small differences between feature sets were observed for other images (data not shown), and hence only Profile LINE features are shown below.

It is very important to select suitable filter parameters because our Gabor kernels respond only to a band of orientations and spatial frequencies (§A.2.2): ill-tuned filters will not respond to major texture structure, and the feature representation may be entirely different when the filter tuning is changed (Ahuja & Rosenfeld, 1981; Marr, 1976). Our artificial “Pop-out” image (Figure 4.5 on page 71) consists of line elements of width one pixel, and consequently contains a range of spatial frequencies. When Gabor filters are tuned to a high spatial-frequency ($T = 3.5$ pixels), the Gabor envelope is tightly localised ($\sigma = 1.1$ pixel) and a small amount of feature “blur” is apparent (Figure 4.21), but this becomes much more noticeable when the filter period is doubled ($T = 7$ pixels; $\sigma = 2.2$ pixels) although this is offset to some extent by the potential for sparse sampling (§4.1.4). These results confirm our prediction (page 89) that apparent contrast decays exponentially either side of the true image feature: $c^2(d) \approx c_0^2 \exp(-d^2/\sigma^2)$. Stripes in the zebra image contain energy at many spatial frequencies, but the sharp peak in the power spectrum confirms that one band predominates (Figure 4.4 on page 68). When filter frequency was set to this value ($T = 7.8$ pixels), the diagonal stripes of the neck were picked out clearly (Figure 4.22a), but the nose was almost invisible; reducing the filter period to $T = 3.5$ pixels gives a more even response (b). For the herringbone

texture (Figure 4.23), a high-frequency filter ($T = 4$ pixels) shown in (b) responds to the fine structure of the material, but at a lower spatial frequency ($T = 8$ pixels), the coarse-scale organisation is much more apparent (c).

Spatial-frequency tuning again influences features extracted from the Westminster image (Figure 4.24): a filter period of 9.5 pixels, suggested by the power-spectrum method, captures much of the ornate structure of the columns and windows but is insensitive to the high spatial-frequency horizontal slats running across the windows. Similarly with the table-mat image, a filter period $T = 32$ pixels picks up the straw loops but is insensitive to the ring structure (Figure 4.25). Spatial frequency tuning is compatible with our objectives (§4.5.1), and allows spatial variation to be recorded for analysis by the structural layer in our hierarchical model. Different characteristics would be more appropriate for forming a “cartoon” of the image, since this seeks to represent “significant” structure at all spatial scales. We have found little effect of increasing the number of orientations at which the image is filtered, usually selecting four ($\Theta = \pi/4$), and ensure that the orientation bandwidth is wide enough to cover the spectrum evenly.

Filter parameters for our final set of feature maps were again selected by the power-spectrum method (§4.3.2); original Brodatz textures are shown in Figure 1.6 on page 11, montages in Figures 4.14 and 4.15 on pages 78 and 80, and feature representations in Figure 4.26 on page 98 (see also Figure 1.5 on page 9). Our features successfully characterise the semi-regular structured textures from which the montages are composed (a–c), but also capture the more intricate patterns shown in (d–f). Previous experiments have shown that Gabor filtering is useful at one level to segment textures into coarse-scale regions with simple energy algorithms (§4.4); this result confirms that they are also suitable at another, for extracting more detailed local structure, and shows that the approximations taken in the derivation of our Profile and Resultant feature-extraction methods do not adversely affect performance.

4.7 Conclusions

Low-level vision demands analysis of both spatial and spectral properties because physical objects are localised yet may be identified reliably only by examining an image neighbourhood. There is a fundamental conflict between these two requirements such that *complete* determination of spatial and spectral properties is not possible, but *partial* accuracy may be achieved along both dimensions according to the uncertainly principle for information (Wilson & Granlund, 1984). Only Gabor kernels allow signals to be measured with minimum joint uncertainty, and hence enjoy unique advantages for image processing as with other forms of communication (Daugman, 1985; Gabor, 1946). In this context, Gabor filtering has been proposed both in its own right in conjunction with texture energy filtering (Chapter 3), and as a model of low-level human visual processing (Chapter 2).

Texture energy models assume that the filtered image varies smoothly within each textured region but sharply at texture boundaries, and we reviewed simple algorithms

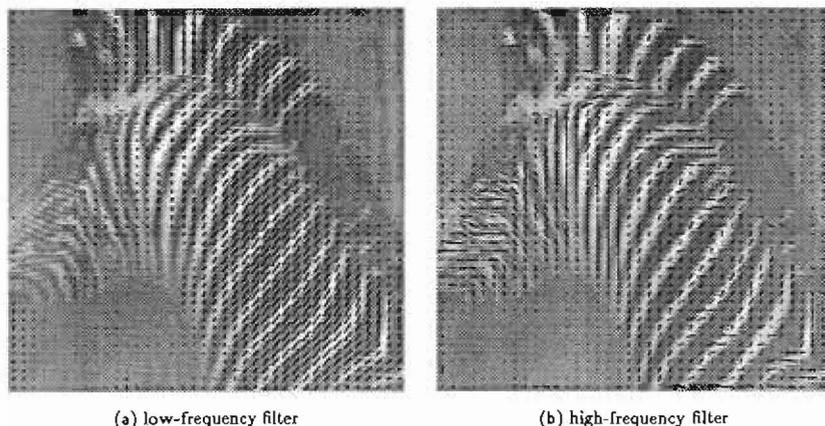


FIGURE 4.22: **Gabor features extracted from zebra image.** Feature vectors were estimated from the zebra image (Figure 4.4 on page 68) using the Profile LINE method, and sampled every four pixels. Needle intensity represents “error”, and length is proportional to “contrast”. (a) low-frequency filter ($T = 7.8$ pixels, $B_{1/2} = 0.5$, $\Delta\theta_{1/2} = \pi/9$, $\Theta = \pi/6$; $\sigma = 8.5$ pixels); (b) high-frequency filter ($T = 3.5$ pixels, $B_{1/2} = 2$, $\Delta\theta_{1/2} = \pi/6$, $\Theta = \pi/4$; $\sigma = 1.1$ pixels), using the notation described on page 69. Filter (a) is tuned selectively to low-frequency stripes in the neck; (b) detects more detailed structure.

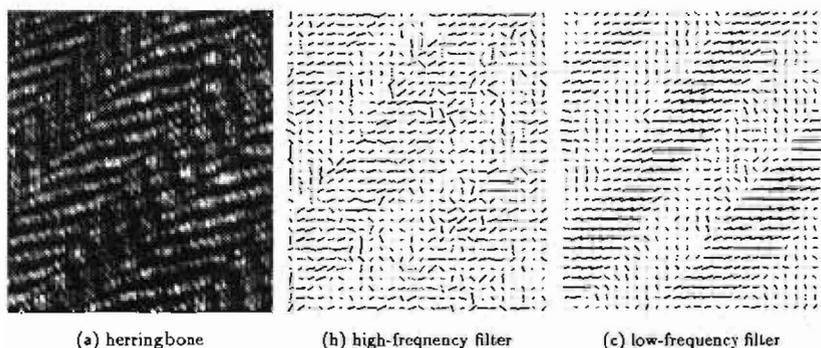
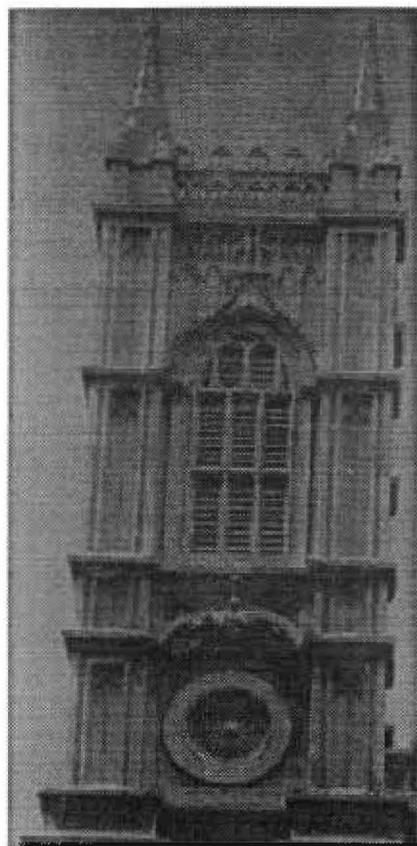
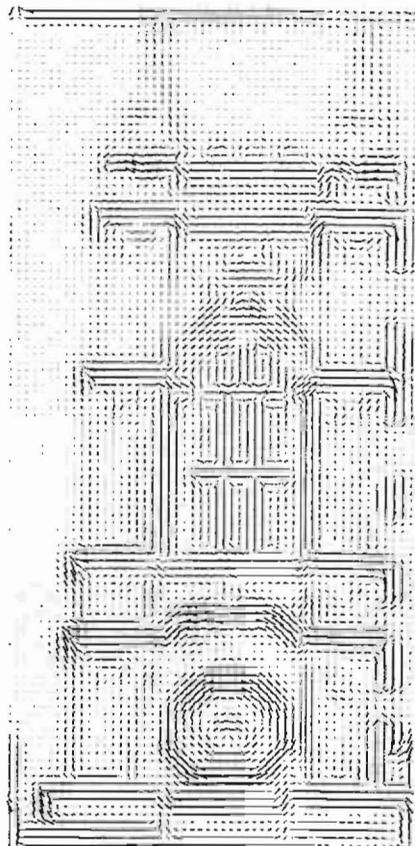


FIGURE 4.23: **Gabor features extracted from herringbone image.** Feature vectors were estimated using the Profile LINE method, and sampled every four pixels. Needle intensity represents “error”, and length is proportional to “contrast”. (a) herringbone texture, 104×122 pixels; (b) high-frequency filter ($T = 4$ pixels; $\sigma = 1.6$ pixels); (c) low-frequency filter ($T = 8$ pixels; $\sigma = 3.1$ pixels). Fine detail is more clear in (b) and the structure of the weave in (c). Other filter parameters: ($B_{1/2} = 1.5$, $\Delta\theta_{1/2} = \pi/6$, $\Theta = \pi/6$), using the notation described on page 69.



(a) Westminister image



(b) Gabor features

FIGURE 4.24: **Gabor features extracted from Westminister image.** Feature vectors were estimated using the Profile LINE method, and sampled every three pixels. (a) Westminister image, 156×328 pixels. (b) Gabor features: needle intensity represents "error", and length is proportional to "contrast". Filter parameters: ($T = 9.5$ pixels, $B_{1/2} = 1.5$, $\Delta\theta_{1/2} = \pi/4$, $\Theta = \pi/6$; $\sigma = 3.7$ pixels), using the notation described on page 69.

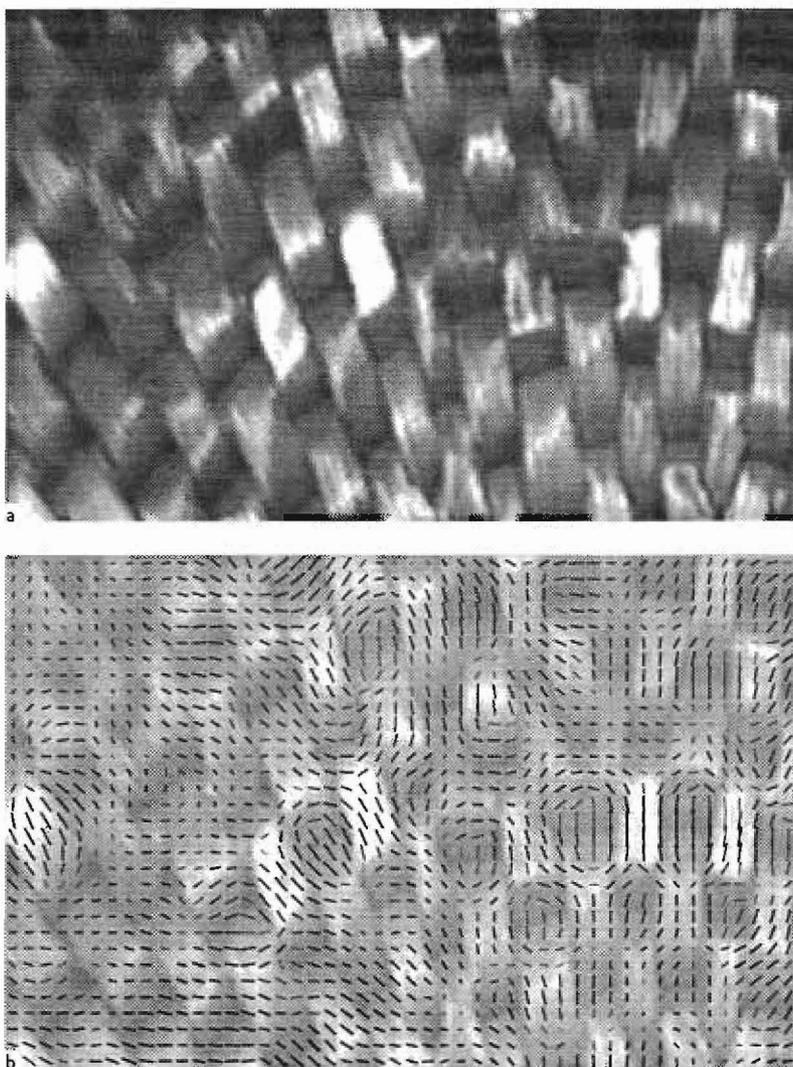


FIGURE 4.25: **Gabor features extracted from mat image.** Feature vectors were estimated by the Profile LINE method, and sampled every five pixels. (a) mat image, 240×160 pixels. (b) Gabor features: needle intensity represents “error”, and length is proportional to “contrast”. Filter parameters: ($T = 32$ pixels, $B_{1/2} = 1$, $\Delta\theta_{1/2} = 2\pi/9$, $\Theta = \pi/4$, $\sigma = 18$ pixels), using the notation described on page 69.

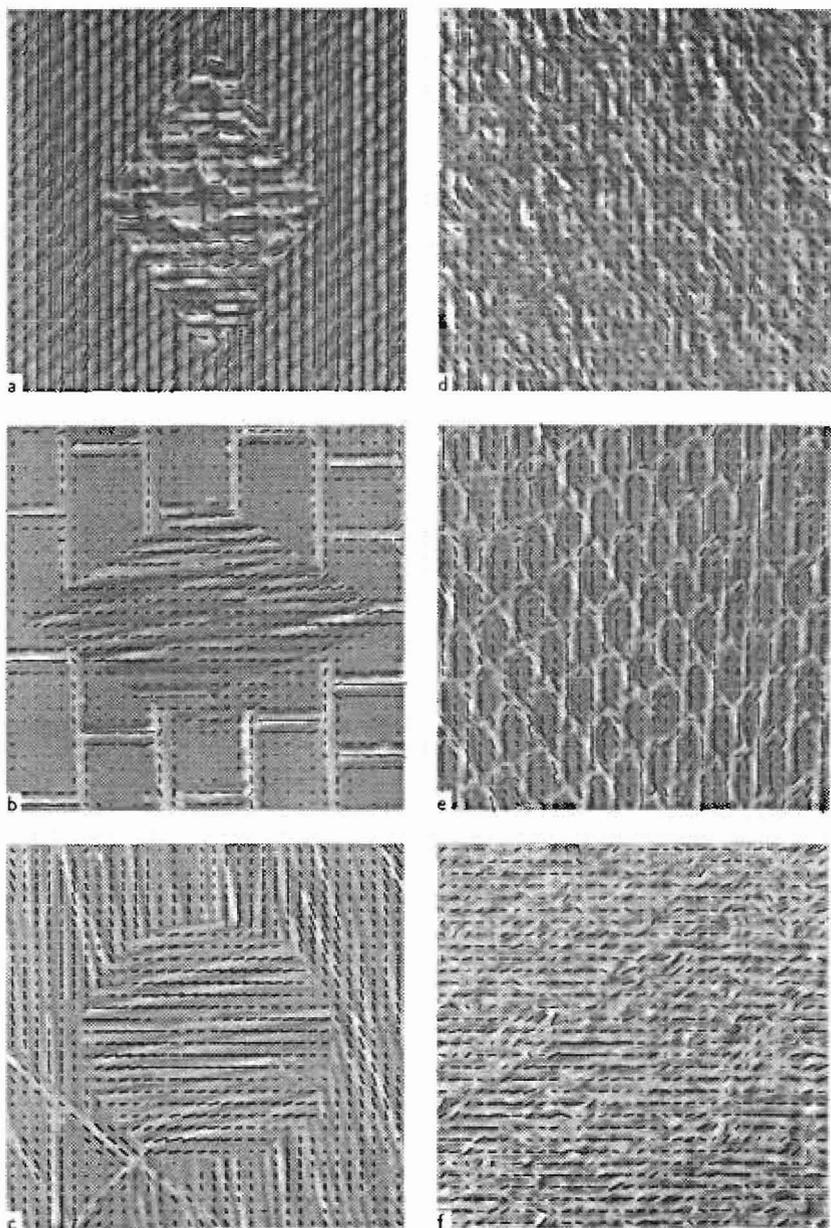


FIGURE 4.26: Gabor features extracted from Brodatz images. (Continued overleaf)

FIGURE 4.26 (*Continued*): Gabor features extracted from Brodatz images. Feature vectors were estimated for these Brodatz montages (Figures 4.14 and 4.15 on pages 78 and 80) and original textures (Figure 1.6 on page 11) using the Profile LINE method with filter parameters suggested by the power-spectrum method (§4.3.2), and sampled every four pixels. Needle intensity represents “error”, and length is proportional to “contrast”. (a) canvas-rafia: ($T = 4.5$ pixels, $B_{1/2} = 0.7$, $\Delta\theta_{1/2} = \pi/6$, $\Theta = \pi/4$; $\sigma = 3.5$ pixels); (b) wall-straw: ($T = 6$ pixels, $B_{1/2} = 1$, $\Delta\theta_{1/2} = 2\pi/9$, $\Theta = \pi/4$; $\sigma = 3.4$ pixels); (c) straw-straw: ($T = 7.3$ pixels, $B_{1/2} = 1$, $\Delta\theta_{1/2} = \pi/6$, $\Theta = \pi/6$; $\sigma = 4.1$ pixels); (d) pressed cork: ($T = 5.5$ pixels, $B_{1/2} = 0.5$, $\Delta\theta_{1/2} = \pi/6$, $\Theta = \pi/4$; $\sigma = 6.0$ pixels); (e) reptile skin: ($T = 6$ pixels, $B_{1/2} = 1$, $\Delta\theta_{1/2} = 2\pi/9$, $\Theta = \pi/4$; $\sigma = 3.4$ pixels); (f) water: ($T = 4.4$ pixels, $B_{1/2} = 1$, $\Delta\theta_{1/2} = \pi/6$, $\Theta = \pi/4$; $\sigma = 2.5$ pixels), using the notation described on page 69.

exploiting this expected behaviour. Our own experiments with artificial textures confirmed some reservations expressed in the literature: variability is often observed *within* each filtered texture, and post-processing used to suppress it has the side-effect of reducing the accuracy with which texture boundaries may be determined. None of the approaches we reviewed could entirely overcome this problem, and we argued that it was unavoidable except for regular textures. Despite these limitations, we found that Gabor energy filtering was sufficient to segment many Brodatz montages both accurately and efficiently, attaining levels of performance qualitatively similar to pre-attentive human vision and offering scope for parallel processing. Performance deteriorated when the image did not obey the severe constraints imposed by the simple texture model, and we identified several circumstances in which these could be violated. These drawbacks, caused by deficiencies in the texture energy algorithms rather than with Gabor filtering *per se*, limit the scope of simple Gabor energy filtering for image analysis.

Accepting that spatial fluctuations in Gabor amplitude arise as unavoidable consequences of texture variability, our approach is to construct hierarchical texture models in which the spatial arrangement of Gabor features is described explicitly. Unlike previous haphazard procedures, we proposed a principled method for combining measurements from several Gabor channels, estimating features by minimising the error between observed spectra and parameterised Gabor signatures. Having considered a number of alternatives, we proposed two efficient algorithms allowing features to be extracted without excessive computational effort. By testing this procedure with real textures, we demonstrated that our feature sets give a sensitive and compact description of texture variability suitable for use with our proposed hierarchical model (Chapter 5).

4.7.1 Summary of Achievements

In this Chapter, we have:

- explored the applicability of Gabor energy filtering for texture segmentation using synthetic and natural textures, and identified causes for observed constraints on performance;

- formed a principled strategy for the extraction of local Gabor features by signature matching;
- proposed efficient algorithms to extract Gabor features using the above procedure, and demonstrated that features obtained from natural textures form a sensitive and compact representation suitable for use with our novel hierarchical framework.

5

Texture Analysis with Gibbs–Markov Random Field Models

Simple spatially-parallel operations are computationally attractive because they may be executed very efficiently by suitable architectures, and it is therefore desirable to employ this paradigm for texture analysis (Chapters 2 and 3). Our experiments with natural and synthetic textures demonstrated the potential of Gabor filtering in this role, and confirmed that simple images are segmented efficiently by algorithms responding to first-order differences in filter output (Chapter 4). Post-processing is often required to refine these rudimentary segmentations, but this can destroy important detail, leading to distorted or missing boundaries, and it is easy to construct textures for which Gabor energy methods fail altogether.

Real textures are inherently variable, but this is not accommodated easily by simple filtering algorithms, which tend to suppress spatial fluctuations. Notions of optimality, hypothesis testing, and random variation are central to statistical study, and hence stochastic models are popular tools for texture analysis. We consider representations employing Gibbs–Markov random fields in this Chapter, which make explicit the probabilistic influence of spatial context within an optimal statistical framework. Many of the image-processing algorithms which apply these models are at least partially serial in nature, and are often computationally intense. This is particularly true for hierarchical random field models, which describe several levels of image structure. Nevertheless, their flexibility and adaptive qualities facilitate automatic segmentation of images containing scenes composed of highly-irregular textures.

In order to establish the statistical framework within which Gibbs–Markov models reside, we first develop the theory of one-dimensional stochastic processes (§5.1).

Auto-normal models are introduced as an important special case whose linear correlation structure greatly simplifies the evaluation of competing statistical hypotheses. We comment on approaches to the estimation of suitable modelling parameters to describe real texture samples (§5.2), and briefly review some applications of Gibbs–Markov models to classic problems in the literature (§§5.3–5.6). Among the deficiencies of current Gibbs–Markov representations are their heavy computational requirement and insufficient level of abstraction. We propose a novel *Gabor–Markov* framework which addresses these concerns by blending the descriptive power offered by Markov random fields with the computational efficiency of Gabor filtering (§5.7). The performance benefits of this new paradigm are demonstrated in Chapter 6.

5.1 Foundations of Gibbs–Markov Random Fields

The probabilistic spatial structure of variable natural textures is made explicit by representations employing Gibbs–Markov random fields, which may operate within an optimal statistical framework. In order to provide a solid foundation for the manipulation and application of Gibbs–Markov texture models (§§5.2–5.6), we establish their theoretical basis in this section. We briefly review properties of one-dimensional stochastic processes and their statistics, in order to establish a theoretical framework (§5.1.1). This extends readily to homogeneous two-dimensional random fields, which describe the behaviour and interaction of lattice sites by probability distributions.

Markov random fields possess a simplified correlation structure, and are specified by the distribution of each lattice site conditional on a collection of other sites, known as its Markov neighbourhood (§5.1.2). The Hammersley–Clifford theorem establishes the formal equivalence between Gibbs and Markov random field models, and the two forms may be used interchangeably (§5.1.3). Statistical manipulation of Markov random field models requires the joint probability distribution of all sites to be computed from the local conditional densities, but this is often computationally arduous, requiring summation over a very large state space. We introduce the special case of auto-normal random fields, whose analysis is greatly simplified because lattice sites interact linearly, and we develop a matrix notation used extensively below (§5.1.5). The reader already familiar with the concept of random fields may prefer to omit §5.1.1, and proceed directly to discussion of the Markov property (§5.1.2) or of auto-normal fields (§5.1.5).

5.1.1 Theory of Stochastic Processes and Fields

A random variable \mathbf{x} assumes a particular value x according to the outcome of a non-deterministic selection procedure. Random quantities will be shown in a different font to distinguish them from ordinary values, as in the previous sentence. The notation $\{\mathbf{x} \leq x\}$ refers to the set of possible experimental outcomes $\{\zeta_i\}$ such that $\mathbf{x}(\zeta_i) \leq x$, and is called an “event”. The probability of each event $\mathcal{P}\{\mathbf{x} \leq x\}$ may be determined by reference to the underlying experimental outcomes (Papoulis, 1990).

A stochastic process represents a small extension to this concept: each outcome ζ , is coupled with a function $x(t, \zeta)$. Single-parameter functions are conventionally described as "time" varying, while two-parameter functions are called "fields", and may be written $\chi(x, y)$ or $\chi(r)$. In both cases, the behaviour of the function is deterministic for a particular experimental outcome ζ , and it may be treated as an ordinary random variable for known parameters t or r . We shall establish the basic results in terms of a one-dimensional process $x(t)$ for the sake of clarity, but extension to two dimensions is obvious.

For a specific t , the process $x(t)$ has distribution function F and density f :

$$\begin{aligned} F(x, t) &= \mathcal{P}\{x(t) \leq x\} \\ f(x, t) &= \frac{\partial F(x, t)}{\partial x} \end{aligned} \quad (5.1)$$

and the link between $F(x, t)$ and $x(t)$ will be made explicit when necessary by writing $F_x(x, t)$ or even $F_{x(t)}(x, t')$. We shall use the notation $\mathcal{P}\{x\}$ for $\mathcal{P}\{x \leq x\}$ where the context is unambiguous.

As we now have more than one random variable, corresponding to different parameter values, their joint (n -th order) statistics are relevant:

$$\begin{aligned} F(x_1, \dots, x_n; t_1, \dots, t_n) &= \mathcal{P}\{x(t_1) \leq x_1, \dots, x(t_n) \leq x_n\} \\ f(x_1, \dots, x_n; t_1, \dots, t_n) &= \frac{\partial^n F}{\partial x_1 \dots \partial x_n}. \end{aligned} \quad (5.2)$$

Statistical Properties of One-Dimensional Processes

The mean and variance of a stochastic process are defined in the usual way, but in general they will be functions of the parameter t :

$$\begin{aligned} \mu_x(t) &= \mathcal{E}\{x(t)\} \\ &= \int_{-\infty}^{\infty} x f(x; t) dx \\ \sigma_x^2(t) &= \mathcal{E}\{x(t)x^*(t)\} - \mu_x(t)\mu_x^*(t). \end{aligned} \quad (5.3)$$

We shall also use the correlation R_{xy} and covariance C_{xy} for two processes $x(t)$ and $y(t)$:

$$\begin{aligned} R_{xy}(t_1, t_2) &= \mathcal{E}\{x(t_1)y^*(t_2)\} \\ &= R_{yx}^*(t_2, t_1) \\ C_{xy}(t_1, t_2) &= R_{xy}(t_1, t_2) - \mu_x(t_1)\mu_y^*(t_2) \end{aligned} \quad (5.4)$$

and because $x(t)$ is a function, its autocorrelation and autocovariance are similarly defined. Note that the autocorrelation $R_{xx}(t, t)$ is non-negative at any instant t :

$$\begin{aligned} R_{xx}(t, t) &= \mathcal{E}\{|x(t)|^2\} \\ &\geq 0. \end{aligned} \quad (5.6)$$

In general, for any function $a(t)$:

$$\begin{aligned} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} a(t_1) R_{xx}(t_1, t_2) a^*(t_2) dt_1 dt_2 &= \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \mathcal{E} \{ a(t_1) x(t_1) a^*(t_2) x^*(t_2) \} dt_1 dt_2 \\ &= \mathcal{E} \left\{ \left| \int_{-\infty}^{\infty} a(t) x(t) dt \right|^2 \right\} \\ &\geq 0 \end{aligned} \quad (5.7)$$

with equality holding only if the functions $x(t)$ are linearly dependent, and hence $R_{xx}(t_i, t_j)$ is non-negative definite. This property has important consequences in the next Chapter.

It is convenient to make use of the *spectral* properties of the process, which are defined in the usual way:

$$X(\omega) = \int_{-\infty}^{\infty} x(t) e^{-j\omega t} dt \quad (5.8)$$

and we shall use the notation $X(\omega) \rightleftharpoons x(t)$ to signify a transform pair. The autocorrelation R_{xx} of a process x may be related to that of its transform, R_{XX} :

$$\begin{aligned} R_{XX}(\omega_1, \omega_2) &= \mathcal{E} \{ X(\omega_1) X^*(\omega_2) \} \\ &= \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} R_{xx}(t_1, t_2) \exp[-j(\omega_1 t_1 - \omega_2 t_2)] dt_1 dt_2 \\ &= \Gamma_{xx}(\omega_1, -\omega_2) \end{aligned} \quad (5.9)$$

where Γ_{xx} is the Fourier transform of R_{xx} : $\Gamma_{xx}(\omega_1, \omega_2) \rightleftharpoons R_{xx}(t_1, t_2)$. The change of sign occurs in (5.9) because of the complex conjugate in the definition of autocorrelation, and our definition for Γ_{xx} follows (Papoulis, 1991) rather than (Papoulis, 1984). This problem does not arise for a real process, when we may write: $R_{XX}(\omega_1, \omega_2) = R_{xx}(t_1, t_2)$.

Stationary and Homogeneous Processes

A stochastic process is *strict-sense stationary* if its statistical properties are invariant to a shift of the parameter origin, implying that the processes $x(t)$ and $x(t+T)$ have identical n -th order distributions:

$$F(x_1, \dots, x_n; t_1, \dots, t_n) = F(x_1, \dots, x_n; t_1 + T, \dots, t_n + T), \quad \forall T. \quad (5.10)$$

In particular, (5.10) implies:

$$f(x, t) = f(x) \quad (5.11)$$

$$f(x_2; t_2 | x_1; t_1) = f(x_2 | x_1; t_2 - t_1), \quad \forall t_1, t_2. \quad (5.12)$$

A homogeneous process may not have a stationary first-order density, and only satisfies the latter relation (5.12). The influence of boundary conditions means that most finite processes are homogeneous rather than stationary.

One consequence of (5-10) is that the time-dependence of the above statistical properties is confined to an *interval* τ :

$$\begin{aligned}\mu_{\mathbf{x}}(t) &= \mu_{\mathbf{x}} \\ \sigma_{\mathbf{x}}^2(t) &= \sigma_{\mathbf{x}}^2 \\ R_{\mathbf{x}\mathbf{y}}(t, t - \tau) &= R_{\mathbf{x}\mathbf{y}}(\tau).\end{aligned}\tag{5-13}$$

A *wide-sense* stationary process satisfies (5-13) but not (5-10).

We shall be concerned with the spectral-density of stationary processes: $S_{\mathbf{x}\mathbf{x}}(\omega) \equiv R_{\mathbf{x}\mathbf{x}}(\tau)$. When \mathbf{x} is stationary, $\Gamma_{\mathbf{x}\mathbf{x}}$ is diagonal:

$$\begin{aligned}\Gamma_{\mathbf{x}\mathbf{x}}(\omega, \omega - \Omega) &\equiv R_{\mathbf{x}\mathbf{x}}(t, t - T) \\ &= \int_{-\infty}^{\infty} R_{\mathbf{x}\mathbf{x}}(T) e^{-j\omega T} dT \int_{-\infty}^{\infty} e^{-j\Omega t} dt \\ &= S_{\mathbf{x}\mathbf{x}}(\omega) \delta(\Omega)\end{aligned}\tag{5-14}$$

implying that $\mathbf{X}(\omega)$ is non-stationary white noise with average power $S_{\mathbf{x}\mathbf{x}}(\omega)$. This conclusion follows because white noise $\mathbf{v}(t)$ is uncorrelated:

$$C_{\mathbf{v}\mathbf{v}}(t_1, t_2) = \sigma_{\mathbf{v}}^2(t_1) \delta(t_1 - t_2).\tag{5-15}$$

If $\mathbf{v}(t)$ is also stationary zero-mean:

$$\begin{aligned}R_{\mathbf{v}\mathbf{v}}(\tau) &= \sigma_{\mathbf{v}}^2 \delta(\tau) \\ S_{\mathbf{v}\mathbf{v}}(\omega) &= \sigma_{\mathbf{v}}^2 \\ R_{\mathbf{N}\mathbf{N}}(\omega) &= \sigma_{\mathbf{v}}^2 \delta(\tau)\end{aligned}\tag{5-16}$$

implying that the transform $\mathbf{N}(\omega) \equiv \mathbf{v}(t)$ is also stationary white noise. In general, the processes defined by (5-15) and (5-16) may assume complex values.

Let us take a stationary process $f(t)$ and transform it by a linear point-spread function $h(t)$ and additive zero-mean stationary white noise $\mathbf{v}(t)$:

$$\begin{aligned}g(t) &= f(t) * h(t) + \mathbf{v}(t) \\ R_{gg}(\tau) &= R_{ff}(\tau) * h(\tau) * h^*(-\tau) + \sigma_{\mathbf{v}}^2 \delta(\tau) \\ S_{gg}(\omega) &= S_{ff}(\omega) |H(\omega)|^2 + S_{\mathbf{v}\mathbf{v}}(\omega)\end{aligned}\tag{5-17}$$

where $H(\omega) \equiv h(t)$. These important results may be obtained by algebraic manipulation (Rosenfeld & Kak, 1982).

Discrete-Time Processes

We shall be concerned mainly with real discrete-time processes in the following Chapters, and it is convenient to represent these using vector (\mathbf{x}) and matrix (\mathbf{R}) notation.

$$\begin{aligned}\mathbf{R}_{\mathbf{x}\mathbf{x}} &= \mathcal{E}\{\mathbf{x}\mathbf{x}^t\} \\ \det \mathbf{R}_{\mathbf{x}\mathbf{x}} &\geq 0\end{aligned}\tag{5-18}$$

If the field \mathbf{x} is homogeneous, its autocorrelation matrix $\mathbf{R}_{\mathbf{x}\mathbf{x}}$ is Toeplitz:

$$R_{\mathbf{x}\mathbf{x}}[k, \ell] = R_{\mathbf{x}\mathbf{x}}[p, q], \text{ if } k - \ell = p - q.\tag{5-19}$$

For a finite sequence \mathbf{x} of length n , the autocorrelation matrix $\mathbf{R}_{\mathbf{x}\mathbf{x}}$ has rank n , and if \mathbf{x} is treated as a segment of a periodic process, $\mathbf{R}_{\mathbf{x}\mathbf{x}}$ is circulant. The discrete Fourier transform may then be taken to obtain $\Gamma_{\mathbf{x}\mathbf{x}} = \mathbf{R}_{\mathbf{x}\mathbf{x}}$, which is diagonal for stationary \mathbf{x} :

$$\Gamma_{\mathbf{x}\mathbf{x}} = \mathcal{F} \mathbf{R}_{\mathbf{x}\mathbf{x}} \mathcal{F}^\dagger \quad (5-20)$$

where \mathcal{F} is the Fourier matrix. A diagonal $\Gamma_{\mathbf{x}\mathbf{x}}$ is much easier to manipulate than $\mathbf{R}_{\mathbf{x}\mathbf{x}}$ directly, and we shall make use of the substitution

$$\mathbf{R}_{\mathbf{x}\mathbf{x}} = \mathcal{F}^\dagger \Gamma_{\mathbf{x}\mathbf{x}} \mathcal{F} \quad (5-21)$$

below. This is particularly convenient because \mathcal{F} is unitary.

Generalisation to Two Dimensions: Random Fields

Results obtained previously in this section for one-dimensional stochastic processes generalise easily to two-dimensional random fields. Correspondence is particularly close for finite random fields, which we shall represent by suitably ordered $M \times N$ vectors and treat as above. The autocorrelation matrix $\mathbf{R}_{\mathbf{x}\mathbf{x}}$ is block-Toeplitz for a homogeneous field \mathbf{x} , and block-circulant for stationary \mathbf{x} . A linear transformation may also be represented as a matrix equation, following (5-17):

$$\begin{aligned} \mathbf{g} &= \mathbf{H}\mathbf{f} + \mathbf{v} \\ \mathbf{R}_{\mathbf{g}\mathbf{g}} &= \mathbf{H}\mathbf{R}_{\mathbf{f}\mathbf{f}}\mathbf{H}^\dagger + \sigma_v^2 \mathbf{1} \\ \Gamma_{\mathbf{g}\mathbf{g}} &= (\mathcal{F}\mathbf{H}\mathcal{F}^\dagger)\Gamma_{\mathbf{f}\mathbf{f}}(\mathcal{F}\mathbf{H}\mathcal{F}^\dagger)^\dagger + \sigma_v^2 \mathbf{1} \end{aligned} \quad (5-22)$$

where \mathbf{H} is a block-Toeplitz matrix of rank MN , \mathbf{f} and \mathbf{g} are the original and transformed fields, respectively, and \mathbf{v} is white noise. Matrix manipulation becomes very computationally demanding for large field size, if not infeasible, but this may be avoided if \mathbf{f} is periodic since $\Gamma_{\mathbf{f}\mathbf{f}}$ is then diagonal. For this reason, it is common to take the approximation of a periodic field by assuming toroidal boundary conditions (Rosenfeld & Kak, 1982).

5.1.2 Markov Random Fields

Natural textures are described conveniently by Markov random field (MRF) models, which represent inherent variability and the influence of spatial context.¹ In this section, we develop MRF theory from the preceding analysis (§5.1.1) by introducing the Markov property, and establish the simplified correlation structure it implies. The values taken by a Markov random field \mathbf{x} defined on a discrete lattice \mathcal{L} at different sites n are conditionally independent provided certain conditions are met. Each element x_n possesses a Markov neighbourhood set $\mathcal{N}_n \subset \mathcal{L}$. If values taken by all sites within this region are known, then x_n is conditionally independent of the remaining surround $\Omega_n = \mathcal{L} \setminus \{n\}$, where $\mathbf{x} \in \mathfrak{S}$.

$$f(x_n | \Omega_n) = f(x_n | \mathcal{N}_n) \quad (5-23)$$

$$\mathcal{E}\{x_n | \Omega_n\} = \mathcal{E}\{x_n | \mathcal{N}_n\} \quad (5-24)$$

¹Despite a comment that "... Markov processes are inherently one-dimensional" (Julesz, 1975, page 38)!

9	8	7	6	7	8	9
8	5	4	3	4	5	8
7	4	2	1	2	4	7
6	3	1	*	1	3	6
7	4	2	1	2	4	7
8	5	4	3	4	5	8
9	8	7	6	7	8	9

FIGURE 5.1: **Markov neighbourhood structure.** A non-causal MRF neighbourhood of order \mathcal{P} contains all sites labelled \mathcal{P} or below. For an auto-normal model (§5.1.5), the neighbourhood coefficients display conjugate symmetry, so only the unshaded part needs to be given. This corresponds to a unilateral neighbourhood. No models of greater than ninth order were used.

Although any neighbourhood set $\mathcal{N}_n \subseteq \Omega_n$ is formally possible, it is usual for it to contain sites symmetrically placed around and immediately adjacent to the “centre” pixel n . We again use a flexible notation here: by $f(\mathbf{x}_n | \mathcal{N}_n)$ we imply $f(\mathbf{x}_n | \mathbf{x}_i = x_i, \forall i \in \mathcal{N}_n)$. The size of the neighbourhood set $\mathcal{N}_{\mathcal{P}}$ is characterised by the *order* of the process, \mathcal{P} . Each successive increment in order adds to \mathcal{N}_n all sites which are equally spaced from and closest to n but which are currently not members, and we shall assume the neighbourhood set takes the same form across the lattice. For a one-dimensional process, the ordering is obvious:

$$\mathcal{N}_{\mathcal{P},n} = \{n + i : 0 < |n - i| \leq \mathcal{P}\} \quad (5-25)$$

and we shall write $\mathcal{N}_{\mathcal{P}} = \mathcal{N}_{\mathcal{P},0}$. The position is a little more complicated in two-dimensions, as illustrated by Figure 5.1: $\mathcal{N}_{\mathcal{P}}$ contains all sites labelled \mathcal{P} or less. A site is always excluded from its own neighbourhood set. When a two-dimensional field is ordered as a vector, adjacent sites are no longer necessarily represented as adjacent elements but this is of no consequence. An MRF of order \mathcal{P} is sometimes referred to as Markov- \mathcal{P} .

When the neighbourhood set is symmetrical, the process is *non-causal* (as in Figure 5.1). If only half the surrounding plane is retained, preserving symmetry along one axis, the process is *unilateral* with neighbourhood set $\mathcal{N}_{\mathcal{P}}^+$, represented by the unshaded part of Figure 5.1. Simplifying further to retain only a quarter plane gives a *causal* process, with neighbourhood $\mathcal{N}_{\mathcal{P}}^c$. It is easier to manipulate models with directed neighbourhood sets, but they are often poor approximations when the structure of the data is genuinely non-causal.

Wide-Sense Markov Processes

The conditional density equation (5-23) defines a strict-sense Markov process. Most properties of interest stem from (5-24), which is obeyed by a wide-sense Markov process when the expected value is replaced by a least-squares estimate $\hat{\mathbf{x}}$:

$$\begin{aligned} \hat{\mathbf{x}}_n &= g_n[\mathcal{N}_n] \\ &= g_n[\Omega_n] \end{aligned} \quad (5-26)$$

represented by the function g_n . As above (5.24), \hat{x} is not affected by the wider surround once neighbouring values are known. The MRF predictor-error u_n is equal to the difference between the estimated and true pixel values, and lies at the heart of Markov models.

$$\begin{aligned} u_n &= x_n - \hat{x}_n \\ \mathcal{E}\{u_n\} &= 0 \end{aligned} \quad (5.27)$$

From the least-squares orthogonality principle,

$$\mathcal{E}\{u_n h(x_m)\} = 0, \quad m \neq n \quad (5.28)$$

where $h(\cdot)$ is an arbitrary function (Papoulis, 1990), and hence the error terms u_n and u_m are correlated only when they lie within a neighbourhood $\mathcal{N}_{\mathcal{P},n}$.

$$\begin{aligned} \mathcal{E}\{x_n u_m^*\} &= R_{uu}(n, n) \delta_{mn} \\ \mathcal{E}\{u_n u_m^*\} &= \begin{cases} R_{uu}(n, m) & m \in \mathcal{N}_{\mathcal{P},n} \\ 0 & m \notin \mathcal{N}_{\mathcal{P},n} \end{cases} \end{aligned} \quad (5.29)$$

From the symmetry of neighbourhood sets, this relation (5.29) is unambiguous:

$$m \in \mathcal{N}_n \iff n \in \mathcal{N}_m. \quad (5.30)$$

The autocorrelation function R_{uu} depends on the particular distribution function of the process x_n .

Homogeneous processes are of particular importance. Since the autocorrelation R_{uu} depends on second-order density, (5.29) may be replaced by:

$$\begin{aligned} \mathcal{E}\{x_n u_m^*\} &= \begin{cases} \sigma_u^2 & n = m \\ 0 & \text{otherwise} \end{cases} \\ \mathcal{E}\{u_n u_m^*\} &= \begin{cases} \sigma_u^2 & n = m \\ R_{uu}(n - m) & n - m \in \mathcal{N}_{\mathcal{P}} \\ 0 & n - m \notin \mathcal{N}_{\mathcal{P}} \end{cases} \end{aligned} \quad (5.31)$$

where σ_u^2 is the MRF variance. We shall be concerned only with homogeneous processes below.

Discrete-State Markov Processes

When the range of the Markov process x_n is restricted to a finite set of values, $a_i \in \mathcal{V}$, the density function may be replaced by a probability vector \mathbf{p} , and the conditional density by a probability matrix $\mathbf{\Pi}$. A one-dimensional discrete-state process is called a Markov chain, and for a homogeneous chain $\mathbf{x}[n]$:

$$\begin{aligned} p_i[n] &= \mathcal{P}\{x[n] = a_i\} \\ \Pi_{i,j}[m] &= \mathcal{P}\{x[n+m] = a_j \mid x[n] = a_i\} \\ \mathbf{\Pi}[n+k] &= \mathbf{\Pi}^{n+k}. \end{aligned} \quad (5.32)$$

If $\mathbf{x}[n]$ is stationary, \mathbf{p} is an eigenvector of $\mathbf{\Pi}$.

$$\mathbf{\Pi}\mathbf{p} = \mathbf{p} \quad (5.33)$$

5.1.3 Gibbs Random Fields

We present the basic definitions of Gibbs distributions in this section, which are relevant because Gibbs and Markov random fields are equivalent. In some circumstances, the Gibbs formulation is more convenient to manipulate, and can take more intuitive parameters.

Gibbs Distributions

A Gibbs distribution $\mathcal{G}(\mathbf{x})$ is an exponential distribution formed from an energy function $U(\mathbf{x})$:

$$\begin{aligned}\mathcal{G}(\mathbf{x}) &= \mathcal{P}\{\mathbf{x} = \mathbf{x}\} \\ &= \frac{1}{Z} e^{-U(\mathbf{x})} \\ Z &= \sum_{\mathbf{x} \in \mathcal{G}} e^{-U(\mathbf{x})}\end{aligned}\tag{5-34}$$

where Z is the normalising constant, or partition function, formed by summing over all possible states $\mathbf{x} \in \mathcal{G}$. The origin of Gibbs distributions lies in physical systems, and it is usual to add a “temperature” parameter, writing:

$$\mathcal{G}(\mathbf{x}, T) = \frac{1}{Z(T)} e^{-U(\mathbf{x})/T}.\tag{5-35}$$

The effect of increasing T is to decrease the distinction between different states, and the physical analogue is clear. As shown in the next section, the energy function $U(\mathbf{x})$ may be defined to construct a Gibbs random field.

Gibbs–Markov Equivalence

The Hammersley–Clifford theorem (Besag, 1974) established the equivalence between Markov and Gibbs random fields. We will summarise this briefly, and introduce a full definition for Gibbs random fields.

We start with a Markov random field \mathbf{x} , assumed homogeneous and discrete-state, defined on a lattice \mathcal{L} of size n : $x_i \in \mathcal{V}$, $\mathcal{G} = \mathcal{V}^n$. The Markov form (5-23) establishes the conditional probability distribution for the i -th pixel, conditioned on its surround \mathcal{N}_i , but we seek the *joint* distribution $P(\mathbf{x})$:

$$\begin{aligned}p_i(x_i) &= \mathcal{P}\{x_i = x_i \mid \Omega_i\} \\ &= \mathcal{P}\{x_i \mid \mathcal{N}_i\}\end{aligned}\tag{5-36}$$

$$\begin{aligned}P(\mathbf{x}) &= \mathcal{P}\{\mathbf{x} = \mathbf{x}\} \\ &= \mathcal{P}\{x_1 = x_1, \dots, x_n = x_n\}.\end{aligned}\tag{5-37}$$

From repeated application of the chain rule (Papoulis, 1990), we may express the relative joint likelihoods of two configurations \mathbf{x} and \mathbf{y} .

$$\frac{P(\mathbf{x})}{P(\mathbf{y})} = \prod_{i \in \mathcal{L}} \frac{\mathcal{P}\{x_i \mid x_1, \dots, x_{i-1}, y_{i+1}, \dots, y_n\}}{\mathcal{P}\{y_i \mid x_1, \dots, x_{i-1}, y_{i+1}, \dots, y_n\}}\tag{5-38}$$

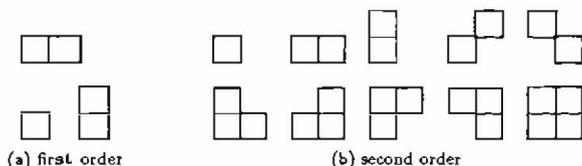


FIGURE 5.2: **Gibbs random field clique types.** A potential function must be defined for each clique. (a) first order; (b) second order. Each clique must be contained within the corresponding neighborhood system (Figure 5.1).

This expression is always valid if $P(y) > 0$ for each combination of possible pixel values $\mathcal{P}\{y_i\} \neq 0$, known as the *positivity* condition. In order to compare different distributions x_i , we may assume an arbitrary “reference” distribution 0 :

$$Q(x) = \ln[P(x)/P(0)].$$

The Hammersley–Clifford theorem (Besag, 1974) gives a unique expansion for $Q(x)$:

$$Q(x) = \sum_{i \in \mathcal{C}} x_i G_i(x_i) + \sum_{\substack{i \in \mathcal{C} \\ j < i}} x_i x_j G_{i,j}(x_i, x_j) + \dots + x_1 \dots x_n G_{1,\dots,n}(x_1, \dots, x_n) \quad (5.39)$$

and the expansion for the conditional distribution $q_i(x_i)$ follows a similar form. The G functions may be chosen arbitrarily, except that $G_S = 0$ unless all members of set S are neighbours, in which case S is a *clique*:

$$G_S \neq 0 \Rightarrow \mathcal{N}_{\mathcal{P}} \cup \{0\} \supseteq \{i - j : i, j \in S\}. \quad (5.40)$$

Valid clique types for first and second-order processes are shown in Figure 5.2. Since each clique must be contained within the extended neighbourhood set $\mathcal{N}_{\mathcal{P},i} \cup \{i\}$, the order of the process influences how many terms of the expansion (5.39) apply.

When the energy function $U(x)$ is formed by summing potentials for each clique, our earlier definition for a Gibbs distribution (5.34) defines a Gibbs random field:

$$U(x) = \sum_{c \in \mathcal{C}} V_c(x) \quad (5.41)$$

where \mathcal{C} is the set of cliques, and $V_c(x)$ is an arbitrary potential function associated with clique c that depends only on the values taken by the members of c . When the field \mathcal{G} is stationary, so are its potentials V_c . In order to specify \mathcal{G} fully, we must give V_c for each possible combination of inputs. Tabulation is manageable for simple processes, but rapidly becomes unwieldy when the number of allowable pixel states is increased, and V_c may be specified as a true function when the process is continuous-state.

Under the Hammersley–Clifford restrictions (5.40) and (5.41), the Gibbs and Markov formulations are equivalent: $\mathcal{G}(x) = P(x)$, although it is possible for one GRF to correspond to many MRFs.² An MRF is established in terms of its local density structure

²We adopt the notation GRF for Gibbs random field, MRF for Markov random field, and GMRF for Gibbs–Markov random field.

whereas a GRF is immediately suggestive of global structure, but normalisation is usually computationally-infeasible in both cases. Summation is required over all possible configurations but this space grows explosively with field size and number of available states.

$$\sum_{\mathbf{x} \in \mathfrak{G}} P(\mathbf{x}) = 1$$

Relative likelihoods of two states may be obtained with much less computational effort, and this is more straightforward with the Gibbs formulation in general, since all that is required is summation of the affected clique potentials.

5.1.4 Auto Models

A restricted class of Gibbs–Markov random fields was proposed by Besag (1974), limiting the Hammersley–Clifford expansion (5.39) to the first two summations and thereby allowing cliques to contain at most two sites.

$$Q(\mathbf{x}) = \sum_{i \in \mathcal{C}} x_i G_i(x_i) + \sum_{\substack{i \in \mathcal{C} \\ j \in \mathcal{C} \\ j < i}} x_i x_j G_{i,j}(x_i, x_j) \quad (5.42)$$

Distributions having this property and an exponential conditional probability distribution are called *auto models*. Under these assumptions, (5.42) may be simplified further:

$$Q(\mathbf{x}) = \sum_{i \in \mathcal{C}} x_i G_i(x_i) + \sum_{\substack{i \in \mathcal{C} \\ j \in \mathcal{C} \\ j < i}} \beta_{i,j} x_i x_j. \quad (5.43)$$

Pixel interaction is linear, corresponding to a linear form for the Markov estimator g_n (5.26), and these equations simplify further if \mathbf{x} is homogeneous. The simplest auto-distribution is auto-binary, since we may write $G_i(x_i) = \alpha_i$. Auto-models are attractive because a linear neighbourhood function makes their analysis tractable, and it also implies that they are only sensitive to first and second-order statistics.

5.1.5 Auto-Normal Models

Both Markov and Gibbs random fields can be difficult to analyse because the expressions involved are often extremely unwieldy. With the Markov formulation, the problem is obtaining an expression for the joint likelihood in terms of the (known) conditional probability distributions because the normalising constant is elusive, requiring summation over a very large state space.

Markov Formulation of Auto-Normal Model

The issue of computational tractability is largely solved by the homogeneous auto-normal model:

$$\begin{aligned} p_i(x_i) &\sim G(\hat{x}_i + \mu, \sigma^2) \\ &= \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left[-\frac{1}{2\sigma^2}(x_i - \mu - \hat{x}_i)^2\right] \end{aligned} \quad (5.44)$$

where $p_i(x_i)$ is the probability of site i conditioned on its surround, as above. Auto-normal models have two key advantages. Firstly, the joint distribution of normal variables may immediately be written down (Papoulis, 1991):

$$f(x_1, \dots, x_N; t_1, \dots, t_N) = \frac{1}{\sqrt{(2\pi)^N |\mathbf{C}|}} \exp \left[-\frac{1}{2} (\mathbf{x} - \boldsymbol{\mu})^\top \mathbf{C}^{-1} (\mathbf{x} - \boldsymbol{\mu}) \right] \quad (5.45)$$

and is fully specified by the mean $\boldsymbol{\mu}$ and covariance matrix \mathbf{C} , assuming a particularly straightforward form when x_i are uncorrelated (\mathbf{C} diagonal). Secondly, the least-squares neighbourhood function g_n (5.26) is linear in the general case (Papoulis, 1991). Any multivariate normal process is therefore auto-normal, and hence a linear neighbourhood function may be assumed without approximation:

$$\hat{x}_n = \sum_{i \in \mathcal{N}_n} \beta_{i,n} x_i \quad (5.46)$$

where $\beta_{i,n}$ are the neighbourhood coefficients, and $\beta_{n,n} = 0$ since each pixel is excluded from its own neighbourhood set. We shall assume that the field \mathbf{x} is homogeneous, and hence may follow (5.31) to write down its correlation structure:

$$x_n = \sum_{i \in \mathcal{N}} \beta_i x_{n+i} + u_n \quad (5.47)$$

$$\mathcal{E} \{x_n u_m^*\} = \begin{cases} \sigma_u^2 & n = m \\ 0 & \text{otherwise} \end{cases}$$

$$\mathcal{E} \{u_n u_m^*\} = \begin{cases} \sigma_u^2 & n = m \\ -\sigma_u^2 \beta_{m-n} & n - m \in \mathcal{N}_p \\ 0 & n - m \notin \mathcal{N}_p \end{cases} \quad (5.48)$$

where u is the MRF predictor error (5.27), and σ_u^2 its variance. We noted above that an autocorrelation function is positive definite (5.6) and hermitian (5.4); and from (5.48) this implies:

$$\det \mathbf{R}_{uu} > 0 \\ \beta_i = \beta_{-i}^* \quad (5.49)$$

or $\beta_i = \beta_{-i}$, for a real process. We shall group these neighbourhood coefficients into a vector $\boldsymbol{\beta}$, and the terms they multiply into $\boldsymbol{\eta}$, so that (5.47) may now be written in vector form:

$$x_n = \boldsymbol{\beta}^\top \boldsymbol{\eta}_n + u_n \quad (5.50)$$

$$\boldsymbol{\eta}_n = \begin{bmatrix} x_{n+1} + x_{n-1} \\ x_{n+2} + x_{n-2} \\ \vdots \end{bmatrix} \quad (5.51)$$

taking as many terms for $\boldsymbol{\eta}_n$ as necessary, but dropping elements for which $\beta_k = 0$. Expressed in this form, only distinct coefficients are contained within $\boldsymbol{\beta}$, since β_i is defined only for $i > 0$.

Finite Auto-Normal Fields

Real applications require the analysis of observed data, for which finite fields are more appropriate. As above (page 106), we shall write the finite two-dimensional field as a vector, using raster-ordering, and express the linear equation (5.47) using matrix algebra:

$$\mathbf{x} = \mathbf{B}\mathbf{x} + \mathbf{u} \quad (5.52)$$

for $\beta_{i,j} = \beta_{i-j}$; alternatively:

$$\mathbf{x} = \mathbf{B}^{-1}\mathbf{u} \quad (5.53)$$

where $\mathbf{B} = \mathbf{1} - \mathbf{B}$. The coefficient matrix \mathbf{B} is hermitian block-Toeplitz, with elements $B_{i,i} = 1$ and $B_{i,j} = -\beta_{i-j}$; and \mathbf{u} is coloured noise. From (5.22) and (5.48):

$$\begin{aligned} \mathbf{R}_{\mathbf{u}\mathbf{u}} &= \sigma_u^2 \mathbf{B} \\ \mathbf{R}_{\mathbf{x}\mathbf{x}} &= \mathbf{B}^{-1} \mathbf{R}_{\mathbf{u}\mathbf{u}} \mathbf{B}^{-1} \\ &= \sigma_u^2 \mathbf{B}^{-1} \end{aligned} \quad (5.54)$$

where $\mathbf{R}_{\mathbf{x}\mathbf{x}}$ and $\mathbf{R}_{\mathbf{u}\mathbf{u}}$ are autocorrelation matrices for the field and predictor-error, respectively. Hence, from (5.45) the joint likelihood $P(\mathbf{x})$ is given by:

$$P(\mathbf{x}) = \sqrt{\frac{|\mathbf{B}|}{(2\pi\sigma^2)^{MN}}} \exp \left[-\frac{(\mathbf{x} - \boldsymbol{\mu})^T \mathbf{B}(\mathbf{x} - \boldsymbol{\mu})}{2\sigma^2} \right] \quad (5.55)$$

where $\boldsymbol{\mu}$ is an arbitrary mean vector, which we shall drop in order to enhance clarity. Note that we also require the coefficient matrix \mathbf{B} to be positive definite because it defines the autocorrelation structure. Inspection of (5.54) shows that the correlation matrix $\mathbf{R}_{\mathbf{x}\mathbf{x}}$ may be written:

$$\mathbf{R}_{\mathbf{x}\mathbf{x}} = \sqrt{\sigma^2 \mathbf{B}^{-1}} \mathbf{1} \left(\sqrt{\sigma^2 \mathbf{B}^{-1}} \right)^T \quad (5.56)$$

implying that the field \mathbf{x} may be expressed as a linear transformation of zero-mean unit-variance white noise \mathbf{v} , since $\mathbf{R}_{\mathbf{v}\mathbf{v}} = \mathbf{1}$:

$$\begin{aligned} \mathbf{x} &= \sqrt{\sigma^2 \mathbf{B}^{-1}} \mathbf{v} + \mathbf{B}^{-1} \mathbf{b} \\ \sqrt{\mathbf{B}} \mathbf{x} &= \sigma \mathbf{v} + \mathbf{b}' \end{aligned} \quad (5.57)$$

where \mathbf{b} and \mathbf{b}' reflect contributions from the boundary conditions. It is therefore possible to obtain a realisation of \mathbf{x} directly once $\sqrt{\mathbf{B}^{-1}}$ is known (Woods, 1972), but the difficulty with this approach lies in forming $\sqrt{\mathbf{B}^{-1}}$ from the known coefficient vector $\boldsymbol{\beta}$ because \mathbf{B} has rank MN (large).

Toroidal Boundary Conditions

A popular approximation with a finite field \mathbf{x} is to assume a periodic extension of the lattice over the plane, and it has only a slight impact for lattices of moderate size despite appearing to be very artificial (Cohen *et al.*, 1991; Chellappa & Kashyap, 1982). Under

these conditions, with \mathbf{x} stationary ($\mu = \mu\mathbf{1}$), the coefficient matrix \mathbf{B} is block-circulant, having a sub-matrix \mathbf{B}_i for each column of the $M \times N$ image:

$$\mathbf{B} = \begin{pmatrix} \mathbf{B}_0 & \mathbf{B}_1 & \cdots & \mathbf{B}_{N-1} \\ \mathbf{B}_{N-1} & \mathbf{B}_0 & \cdots & \mathbf{B}_{N-2} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{B}_1 & \mathbf{B}_2 & \cdots & \mathbf{B}_0 \end{pmatrix} \quad (5-58)$$

where \mathbf{B}_i are circulant and have rank M . Following (5-21), \mathbf{B} may be diagonalised by the Fourier matrix:

$$\mathbf{B} = \mathcal{F}^T \mathbf{A} \mathcal{F} \quad (5-59)$$

$$\mathbf{A} = \Gamma_{uu} / \sigma_u^2 \quad (5-60)$$

where \mathbf{A} is diagonal with elements $\lambda_i = S_{uu,i} / \sigma_u^2$, and has rank MN .

$$\lambda_{m+nM} = 1 - 2 \sum_{(k,\ell) \in \mathcal{N}^+} \beta_{k+\ell S} \cos \left[2\pi \left(\frac{mk}{M} + \frac{n\ell}{N} \right) \right] \quad (5-61)$$

The double subscripts on λ_{m+nM} and $\beta_{k+\ell S}$ reflect raster-ordering of the array: the field lattice \mathcal{L} has width M , and the causal neighbourhood \mathcal{N}^c has span S . Where the context is obvious, we shall instead write λ_i or β_k , and use the two forms interchangeably below.³ Effectively, the finite relation (5-57) has been re-written as a convolution.

$$\mathbf{x} * (\delta - \beta) = \mathbf{u}$$

We shall also write

$$\lambda_i = 1 - B_i \quad (5-62)$$

noting from (5-61) that \mathbf{B} is the cosine transform of β :⁴

$$B_{m+nM} = 2 \sum_{(k,\ell) \in \mathcal{N}^+} \beta_{k+\ell S} \cos \left[2\pi \left(\frac{mk}{M} + \frac{n\ell}{N} \right) \right]. \quad (5-63)$$

Using the diagonal form for \mathbf{B} , and noting that: $|\mathcal{F}^T \mathbf{A} \mathcal{F}| = \prod_i \lambda_i$; we may write down the joint likelihood of a periodic field \mathbf{x} :

$$\begin{aligned} P(\mathbf{x}) &= \sqrt{\left[\frac{|\mathcal{F}^T \mathbf{A} \mathcal{F}|}{(2\pi\sigma_u^2)^{MN}} \right]} \exp\left(-\frac{\mathbf{x}^T \mathcal{F}^T \mathbf{A} \mathcal{F} \mathbf{x}}{2\sigma_u^2}\right) \\ &= \prod_{i \in \mathcal{L}} \left(\frac{\lambda_i}{2\pi\sigma_u^2} \right)^{1/2} \exp\left(-\frac{|X_i|^2 \lambda_i}{2\sigma_u^2}\right) \end{aligned} \quad (5-64)$$

$$2L(\mathbf{x}) = \sum_{i \in \mathcal{L}} \log \lambda_i - MN \log(2\pi\sigma_u^2) - \sum_{i \in \mathcal{L}} \frac{|X_i|^2 \lambda_i}{\sigma_u^2} \quad (5-65)$$

³Spatial and spectral lattice co-ordinates are not equivalent as we have implied, and we introduce the "spectral mapping function" in §7.2.

⁴Note that \mathbf{B} is a coefficient matrix derived from the vector β (5-53), and that \mathbf{B} is the cosine transform of β , with elements B_i (5-63). We shall not refer again to individual elements of \mathbf{B} .

where $\mathbf{X} = \mathcal{F}\mathbf{x}$, summation for i extends over the image lattice \mathcal{L} , and $L(\mathbf{x}) = \log P(\mathbf{x})$. Symmetry of $\mathbf{R}_{\mathbf{x}\mathbf{x}}$ is reflected by λ_i being real; positive-definiteness implies λ_i positive:

$$\lambda_i > 0, \forall i \in \mathcal{L}. \quad (5.66)$$

This condition on the neighbourhood coefficient vector β is clearly satisfied by:

$$\sum_{i \in \mathcal{N}^+} \beta_i (z^i + z^{-i}) < 1, \forall z \in \mathbb{C} : |z| = 1 \quad (5.67)$$

but this is more restrictive than necessary, and (5.66) suffices for finite fields. Whether β represents a valid stationary field depends on the range of summation $i \in \mathcal{L}$, and hence on field size. The relation (5.66) is universal, and since it was derived under the assumption that the field was homogeneous, we shall name it the *homogeneity constraint*. It has important consequences for MRF parameter-estimation methods, since approximations may lead to invalid parameter sets which violate this constraint (§5.2).

Ignoring boundary conditions, the relation (5.57) may now be written:

$$\mathbf{x} = \sigma \mathcal{F}^t \sqrt{\Lambda^{-1}} \mathcal{F} \mathbf{v} \quad (5.68)$$

but the Fourier transform of unit-variance zero-mean white noise \mathbf{v} is unit-variance zero-mean white noise (5.16), giving the distribution for \mathbf{X} :

$$X_i \sim G(0, \sigma^2 / \lambda_i). \quad (5.69)$$

The field \mathbf{x} is usually real, and hence its complex spectrum \mathbf{X} displays conjugate symmetry.

Causal Representation

An alternative simplification is available for finite linear fields by rewriting (5.52) as:

$$\mathbf{x} = \mathbf{L}\mathbf{x} + \mathbf{v} \quad (5.70)$$

where \mathbf{L} is lower-triangular with zero leading diagonal (Stuller & Kurz, 1976). This is a causal representation, which often makes analysis easier. For a Markov- \mathcal{P} field and a lattice of size $N \times N$, the order of the new field is $N\mathcal{P}$ — this may be very large in practice, and we prefer to take advantage of the toroidal boundary approximation. A causal process is driven by white noise \mathbf{v} :

$$x_n = \sum_{i \in \mathcal{N}^c} \beta_i x_{n-i} + v_n \quad (5.71)$$

where \mathcal{N}^c denotes the causal neighbourhood.

Gibbs Formulation of Auto-Normal Model

Following the Hammersley–Clifford theorem (§5.1.3), the auto-normal MRF may also be expressed as a Gibbs random field. From (5.43), this takes a particularly simple form: cliques contain at most two pixels, reflecting the linear neighbourhood function. The weighting given to single-pixel cliques, $G_i(x_i)$, determines the normal form.

5.1.6 Summary: Properties of Gibbs–Markov Random Fields

Markov random fields are stochastic processes having a simplified autocorrelation structure. In this section, we have introduced the properties and definitions that are of use below in the application of Gibbs–Markov random fields to image analysis (§§5.2–5.6). Under certain conditions, Gibbs and Markov random fields are equivalent (§5.1.3), and either or both forms may be used in a particular application. Determination of the joint likelihood of all lattice sites is computationally arduous in general, but is greatly simplified by the use of auto-models, and by the auto-normal model in particular. Manipulation of auto-normal models is often far more convenient than other Gibbs–Markov forms because pixels interact linearly, and there is no tractable alternative in some cases. This simplification has great practical significance, and it is common to adopt auto-normal models for practical applications.

5.2 Estimation of Parameters for Gibbs–Markov Models

If Gibbs–Markov random fields are to be used as models of *natural* textures, it is essential to be able to estimate a parameter set to characterise each texture type given suitable sample images. Optimal parameter estimates are obtained by maximising the likelihood of the observed training data with respect to the parameterised Gibbs–Markov model, but this is often infeasible because the joint likelihood of all image pixels can not be found in closed form (§5.2.1). Code and pseudo-likelihood estimation methods substitute an approximate function that can usually be maximised with only moderate computational effort, leading to acceptable but sub-optimal parameter estimates (§5.2.2; §5.2.3). Heuristic approximations are also required to estimate parameters for the Gibbs formulation (§5.2.4). None of these restrictions applies to auto-normal models, whose simplified correlation structure leads to a closed form for the joint likelihood (§5.2.5). This must still be maximised numerically, but pseudo-likelihood estimates are available analytically and are sometimes preferred.

Situations do arise where no suitable training data is available: modelling scene layout is a common example. Without particular prior knowledge, the requirement is to describe how an arbitrary image is partitioned between different textured regions, and a general assumption of spatial coherence is often used to bias the model in favour of homogeneous clumps of texture. Attempting to estimate a parameter set is inappropriate in these circumstances, but it is straightforward to design a Gibbs random field with the correct form (Derin & Cole, 1986; Geman & Geman, 1984; Hansen & Elliott, 1982).

5.2.1 MRF True-Likelihood Estimation

Let us assume that we have an MRF with a given conditional density distribution and neighbourhood size, and wish to select a parameter set that most closely matches some training data. Maximum-likelihood estimation takes a realisation \mathbf{x} and model \mathfrak{X} , and

seeks to maximise their joint likelihood $\mathcal{P}\{\mathbf{x}, \mathfrak{X}\}$ or log-likelihood $L(\mathbf{x}, \mathfrak{X})$. Suppose $\mathfrak{X} = \mathfrak{X}(\boldsymbol{\theta})$ where $\boldsymbol{\theta} = \{\theta_i\}_{i=1}^n$ are the parameters to be estimated:

$$\begin{aligned} \left. \frac{\partial L}{\partial \theta_i} \right|_{\theta_i = \hat{\theta}_i} &= 0, \quad i = 1..n \\ \hat{\mathfrak{X}} &= \mathfrak{X}(\hat{\boldsymbol{\theta}}) \end{aligned} \quad (5.72)$$

where $\hat{\boldsymbol{\theta}}$ is the true-likelihood parameter set. When several realisations $\{\mathbf{x}_i\}$ are available, their joint likelihood is maximised. Unfortunately, the joint likelihood of a field $P(\mathbf{x})$ is difficult to compute in general, for the reasons outlined in §5.1.3, and computation of true maximum-likelihood estimates is therefore often infeasible in practice.

5.2.2 MRF Code Estimation

An alternative approximate method is available because the conditional densities $p_i(x_i)$ are known. If the sites were independent, we could easily combine conditional likelihoods to form the joint likelihood:

$$\begin{aligned} P_k(\mathbf{x}) &= \prod_{i \in Q(k)} p_i(x_i) \\ L_k(\mathbf{x}) &= \ln P_k(\mathbf{x}) \\ &= \sum_{i \in Q(k)} \ln p_i(x_i) \end{aligned} \quad (5.73)$$

$$Q_k = \{i \in \mathcal{L} : Q_k \cap \mathcal{N}_i = \emptyset\} \quad (5.74)$$

where \mathbf{x} is defined on the lattice \mathcal{L} . From the Markov assumption, sites are only correlated out to a certain distance, and hence the joint distribution of well-separated sites may indeed be written as (5.73). This approach is called *coding* (Besag, 1974); each set of well-separated elements Q_k is a code. In effect, we are maximising the joint likelihood for each code conditional on the rest of the image. It is usual, but not essential, to make each code as large as possible within the constraint (5.74) so that each estimate is as efficient as possible, and minimising the number of codes κ required to cover the lattice.

$$\begin{aligned} \mathcal{L} &= \bigcup_{i=1}^{\kappa} Q_i \\ Q_i \cap Q_j &= \emptyset, \quad \forall i \neq j \end{aligned} \quad (5.75)$$

Coding is an effective way of overcoming the intractable partition function required by true-likelihood estimation, but there seems to be no sensible way of combining parameters from each code other than an arithmetic average. This is unfortunate because they are not independent, but they are often numerically similar. Coding makes rather an inefficient use of the training data because each code uses only a subset of the field, and this problem becomes particularly acute for fields of high MRF order. The coding method does not depend on a particular form for the conditional distribution function $p_i(x_i)$, and has been successfully applied with several model variants (Besag,

1974; Cross & Jain, 1983). Code likelihoods L_k (5-73) must be maximised numerically in general, and there is unfortunately no guarantee that the final parameter set will be consistent with respect to the homogeneity constraint (5-66).

5.2.3 MRF Pseudo-Likelihood Estimation

The chief disadvantage of the coding method is the rather inefficient use of training data, particularly for fields of large order. For example, third-order neighbourhoods contain twelve sites, so each code uses only $7\frac{1}{2}\%$ of the data directly. A simple way of increasing this is to extend the range of summation in (5-74) to *all* pixels of a given texture type (other than boundary pixels, if any):

$$\begin{aligned}\tilde{P}(\mathbf{x}) &= \prod_{i \in \mathcal{L}} p_i(\mathbf{x}_i) \\ \tilde{L}(\mathbf{x}) &= \sum_{i \in \mathcal{L}} \ln p_i(\mathbf{x}_i)\end{aligned}\quad (5-76)$$

where \mathcal{L} is the entire image lattice. The quantities $\tilde{P}(\mathbf{x})$ and $\tilde{L}(\mathbf{x})$ are not true likelihood functions but the product of conditional likelihoods, and lead to approximate parameter estimates, $\tilde{\theta}$ or $\tilde{\mathbf{B}}$. Advantages of the pseudo-likelihood are: it is easy to compute, even over an irregularly-shaped region; it is more efficient than the coding method; and it generates acceptable parameter estimates in practice (Besag, 1986; Cohen & Cooper, 1987; Derin & Elliott, 1987). It is usual for more questionable approximations to be taken at other stages of image analysis, and hence sub-optimal pseudo-likelihood parameter estimates may be acceptable. They do suffer from MRF consistency problems because the homogeneity constraint (5-66) is not always observed, but these may sometimes be compensated for (Cohen & Cooper, 1987), and pseudo-likelihood estimation is popular because of its computational simplicity.

5.2.4 GRF Histogram Parameter-Estimation Method

The above methods estimate Markov parameters, and suffer from the disadvantage that the estimated parameters may not obey the restrictions for a valid MRF: the field is often assumed to be stationary, but the homogeneity constraint (5-66) may be violated. This consistency problem is avoided by using the alternative Gibbs formulation, but true maximum-likelihood estimation is no more tractable than above (§5.2.1) because of the difficulty in calculating the normalising constant Z (5-34).

The "histogram method" is an approximation to true maximum-likelihood that overcomes this practical difficulty (Derin & Elliott, 1987). We define V_i to be the sum of potential functions for all cliques c containing the pixel i :

$$V_i = \sum_{c \in \mathcal{C}: i \in c} V_c(\mathbf{x}) \quad (5-77)$$

where V_i may be written as a product between the (column) parameter vector θ and a suitable neighbourhood function ϕ .

$$V_i = \phi^T \theta \quad (5-78)$$

The number of terms in ϕ and θ depends on the symmetry and the order of the GRF, but auto-models have fewer parameters than general fields (§5.1.4), and are suitable for use with this method. From the Gibbs-Markov equivalence (§5.1.3), it follows that:

$$\begin{aligned} \mathcal{P}\{\mathbf{x}, = \mathbf{x} \mid \Omega_i\} &= \mathcal{P}\{\mathbf{x} \mid \mathcal{N}_i\} \\ &= \frac{\mathcal{P}\{\mathbf{x}, \mathcal{N}_i\}}{\mathcal{P}\{\mathcal{N}_i\}} \\ &\propto \exp[-V_i(\mathbf{x}, \mathcal{N}_i)] \end{aligned} \quad (5-79)$$

$$\begin{aligned} \ln \left[\frac{\mathcal{P}\{\mathbf{x}_1 \mid \mathcal{N}_i = \mathcal{X}_i\}}{\mathcal{P}\{\mathbf{x}_2 \mid \mathcal{N}_i = \mathcal{X}_i\}} \right] &= -V_i(\mathbf{x}_1, \mathcal{X}_i) + V_i(\mathbf{x}_2, \mathcal{X}_i) \\ &= [\phi(\mathbf{x}_2, \mathcal{X}_i) - \phi(\mathbf{x}_1, \mathcal{X}_i)]^T \theta. \end{aligned} \quad (5-80)$$

where \mathcal{N} is the Markov neighbourhood. If the LHS of (5-80) is known, this vector relation may be solved as linear equations in the unknown parameters θ and neighbourhood function ϕ . In general, θ is over specified provided the number of gray-levels is small, the cliques have low order, or simplifying assumptions are made, and least-squares solutions may be sought. The probabilities $\mathcal{P}\{\mathbf{x} \mid \mathcal{N}\}$ may be estimated by forming a histogram for all observed combinations of neighbourhood and centre pixel. The law of large numbers:

$$\mathcal{P}\{A\} \simeq n_A/n, \quad n \text{ large}$$

may be invoked to form the estimates. A large amount of trial data is needed to estimate these probabilities accurately for all pixel combinations, particularly for fields of moderate order or for many pixel values, when the number of permutations of \mathbf{x}, \mathcal{N} is large. However, since the desired parameter vector θ is usually over-specified, it is not necessary to evaluate every term. An advantage of the histogram method is that it is free of consistency problems, and is straightforward to implement. A disadvantage is that it requires a large amount of training data, although this is also true of the coding method, and its formal properties have not been thoroughly analysed.

5.2.5 Estimation of Auto-Normal Parameters

True-Likelihood Parameter Estimation

For the general cases considered above, it was necessary to use approximate methods to locate the best parameter estimates because the joint likelihood $P(\mathbf{x})$ was unavailable (§§5.2.2-5.2.4). This is no longer an obstacle for homogeneous auto-normal models, when $P(\mathbf{x})$ may be evaluated comparatively easily (§5.1.5).

True-likelihood estimation demands maximisation of $L(\mathbf{x}, \mathfrak{X})$ with respect to the parameter set $\mathfrak{X} = \{\mu, \sigma^2, \beta\}$, where the neighbourhood coefficient vector β also determines the coefficient matrix \mathbf{B} . For an image of size n , from (5-55) and (5-72):

$$\hat{\sigma}^2 = \frac{\mathbf{x}^T \hat{\mathbf{B}} \mathbf{x}}{n} \quad (5-81)$$

and the maximum likelihood estimate $\hat{\mathbf{B}}$ minimises

$$n \ln(\mathbf{x}^T \mathbf{B} \mathbf{x}) - \ln |\mathbf{B}|$$

from which $\hat{\mathbf{B}}$ and hence $\hat{\beta}$ may be obtained by Newton–Raphson iteration (Besag, 1974). The toroidal boundary approximation allows \mathbf{B} to be diagonalised (5.64), and the estimate to be found more easily, but a closed-form solution for $\hat{\beta}$ does not exist in general (Kashyap & Chellappa, 1983). In practice, the requirement for numerical optimisation is a drawback of true-likelihood estimation — it has even been described as “cumbersome and unreliable” (Derin & Elliott, 1987). Approximate closed-form solutions are free from this uncertainty.

Pseudo-Likelihood Parameter Estimation

True-likelihood methods require numerical techniques to locate the best parameter estimate, and this may require more CPU time than is available. A popular alternative is pseudo-likelihood estimation (§5.2.3), which obtains inferior parameter estimates but requires a predictable and much-reduced amount of computation, and is usually preferred to the coding method (§5.2.2).

From definitions for the local conditional probability distribution (5.44) and pseudo-likelihood function (5.76), and dropping the mean:

$$2\tilde{L}(\mathbf{x}) = MN \log(2\pi\sigma^2) - \sum_{i \in \mathcal{L}} \frac{(x_i - \beta^\top \boldsymbol{\eta}_i)^2}{\sigma^2} \quad (5.82)$$

where the image is defined on a lattice \mathcal{L} of size $M \times N$. Maximisation according to (5.72) gives:

$$\begin{aligned} \hat{\beta} &= \left[\sum_{i \in \mathcal{L}} \boldsymbol{\eta}_i \boldsymbol{\eta}_i^\top \right]^{-1} \sum_{i \in \mathcal{L}} \boldsymbol{\eta}_i x_i \\ \hat{\sigma}^2 &= \frac{1}{MN-1} \sum_{i \in \mathcal{L}} \left(x_i - \hat{\beta}^\top \boldsymbol{\eta}_i \right)^2 \end{aligned} \quad (5.83)$$

where $\boldsymbol{\eta}$ is the neighbourhood vector (5.51). Because $\tilde{L}(\mathbf{x})$ is not a true likelihood, there is a danger that the interaction matrix $\hat{\mathbf{B}}$ estimated from it will not be positive definite, and hence that the homogeneity constraint (5.66) will not always hold. Whether this is a serious deficiency clearly depends on the precise application of the estimated parameter set, but true-likelihood methods are preferred if parameter estimation may be performed off-line. In this case, the pseudo-likelihood function (5.82) may still be used in subsequent analysis. For auto-normal models, pseudo-likelihood estimation is asymptotically more efficient than coding estimates (Kashyap & Chellappa, 1983).

5.2.6 Verification of Parameter Sets

Having obtained a parameter estimate, it is often desirable to test whether it describes the training data adequately (§3.6.1). Discrepancies may arise because of approximation errors in estimation, because of unjustified assumptions made about the field, or simply because of the variation inherent in a stochastic process. One simple and effective but subjective verification method is to reconstruct an image with the estimated parameters

and to compare it visually with the original (Cross & Jain, 1983; Kashyap *et al.*, 1982). Field synthesis methods are discussed in §5.3.

A more scientific test may be made by forming a goodness-of-fit hypothesis. When the number of configurations is reasonably small, the Pearson statistic may be used, leading to a χ^2 -test (Besag, 1974; Cross & Jain, 1983). This involves comparing actual and expected frequencies for each neighbourhood situation, and only gives reliable results if the observations are not sparse. When the pixel values are (almost) continuous and normal, analysis of variance leading to an F -test is more appropriate (Besag, 1974).

An approximate Bayesian statistic has been proposed to select the correct neighbourhood structure, providing a more objective and quantitative test than visual inspection (Kashyap & Chellappa, 1983). Selection of the best neighbourhood is complicated because a model with more parameters always has greater freedom to model the data more closely, and residual error is always a decreasing function of model order.

5.3 Texture Synthesis from Gibbs–Markov Models

An attractive qualitative method of demonstrating that important texture characteristics are represented by a measured parameter set is to generate an artificial texture from it (§5.2.6). A discussion of approaches to texture synthesis allows us to introduce some generic techniques for manipulating Gibbs–Markov models, which have also been used for more concrete applications (§§5.5–5.6). Ideally, field synthesis is straightforward: all that is required is to sample the joint distribution. Complications arise because the joint likelihood is generally unavailable, with the exception of homogeneous auto-normal fields when synthesis is indeed straightforward (§5.3.5). Usually, only the local conditional density $p_i(x_i)$ is known, either directly from the MRF formulation, or from summing and comparing local GRF clique potentials.

Probabilistic methods may be used to move from local conditional to joint distribution, and have wider application than field synthesis (§§5.5–5.6). Performance of this technique is not guaranteed for a particular realisation because of its non-deterministic nature, but it can be shown that the expected result has the correct form. Relaxation algorithms seek to refine the initial state iteratively until its distribution takes the desired form, and are usually computationally demanding. Some approaches to parallel implementation are discussed in §5.6.6.

5.3.1 Monte Carlo Algorithm

In essence, a Monte Carlo (or Metropolis) algorithm assumes the field takes state $\mathcal{S}(t)$ and perturbs it to generate $\mathcal{S}(t+1)$ in such a way that as $t \rightarrow \infty$ the distribution of \mathcal{S} tends to that of the required joint distribution. Each perturbation involves changing the state of at most two sites, and hence computation of local conditional probabilities is sufficient to calculate the change in the joint distribution. Each time step has associated with it a transition \mathcal{T} , which may be either accepted or rejected according to the change in joint likelihood it effects. With the “flip” algorithm, the state of a single site is set to a new value; for the “exchange” variant, a pair of sites may be exchanged. Clearly,

the first-order distribution is unaffected by the exchange procedure, freezing in pixel values in the same proportion as the initial state. It is usually desirable to minimise the importance of the starting state, because this may have been chosen in an *ad hoc* manner.

At the core of the algorithm lies the selection of which transitions are allowed to proceed.

$$\begin{aligned}\mathcal{T} &= \mathbf{x} \mapsto \mathbf{x}' \\ \rho_{\mathcal{T}} &= P(\mathbf{x}')/P(\mathbf{x})\end{aligned}\quad (5.84)$$

For simplicity, let us assume a “flip” model, with a possible transition \mathcal{T} at site i . Both i and \mathcal{T} should be generated to ensure possible field configurations are sampled evenly. Relative likelihoods of the two global states \mathbf{x} and \mathbf{x}' are obtained from known conditional densities according to the Hammersley–Clifford theorem (5.38), and summation need only extend over those sites whose neighbourhood contains i :

$$\rho_{\mathcal{T},\text{MARF}} = \frac{p_i(\mathbf{x}'_i)}{p_i(\mathbf{x}_i)} \prod_{k \in \mathcal{L}: i \in \mathcal{N}_k} \frac{\mathcal{P}\{\mathbf{x}_k | \mathcal{X}_k, \mathbf{x}'_i\}}{\mathcal{P}\{\mathbf{x}_k | \mathcal{X}_k, \mathbf{x}_i\}}.$$

Similarly, $\rho_{\mathcal{T},\text{GRF}}$ is computed by summing potentials V_i for all cliques containing i , since other cliques do not change.

$$\rho_{\mathcal{T},\text{GRF}} = \exp[V_i(\mathbf{x}_i, \mathcal{X}_i) - V_i(\mathbf{x}'_i, \mathcal{X}_i)]$$

As with deterministic greedy algorithms, the transition \mathcal{T} is always accepted when $\rho_{\mathcal{T}} \geq 1$, but it may also be accepted when it causes a *decrease* in the joint likelihood.

$$\text{ACTION} \triangleq \begin{cases} \rho_{\mathcal{T}} \geq 1 & \text{ACCEPT} \\ \rho_{\mathcal{T}} < 1 & \text{ACCEPT with probability } \rho_{\mathcal{T}} \end{cases} \quad (5.85)$$

“Convergence” may be determined by monitoring the statistics of the field \mathbf{x} compared with their anticipated values, or by counting the proportion of transitions which are accepted. There is no clear end-point because equilibrium around the final state is dynamic, as the algorithm continually samples the required distribution. The sequence of states \mathcal{S} , is a Markov chain (§5.1.2), whose ergodicity allows a proof that the synthesised field \mathbf{x} has a distribution corresponding to the limiting state of the chain, which is the eigenvector of its transition matrix (5.33). This distribution is only approached once transients caused by the choice of initial state have died away.

As described here, the algorithm operates at a single resolution and hence changes in state propagate across the field slowly. Recovery from a poor choice of initial state may take many iterations although the limiting distribution is unaffected. Only a single transition is considered at each iteration, and so this algorithm is not very suitable when each pixel may assume many possible values. It has been used successfully with the binomial distribution for up to 32 gray levels although this required many minutes’ CPU time per image (Cross & Jain, 1983).

5.3.2 Gibbs Sampler

Another stochastic relaxation algorithm closely related to the above is the "Gibbs sampler" (Geman & Geman, 1984). Transitions are not restricted to fixed candidates but are generated directly from the local conditional distribution at each iteration. Let us assume a homogeneous field \mathbf{x} defined on a lattice \mathcal{L} of size n , for $\mathbf{x} \in \mathfrak{S} = \mathcal{V}^n$. At each time pulse t the state of a single site x_i may be updated, and its proposed new state x_i' is drawn directly from the conditional density distribution $p_i(x_i)$, conditioned on the unchanged surround \mathcal{X}_i . This process is repeated across the lattice using any sensible ordering (which need not be deterministic). Unlike the Monte Carlo algorithm, the Gibbs sampler considers *all* possible values at site i and disregards the present state.

Much of the novelty of the Gibbs sampler lies in the use of the "temperature" parameter T , which effectively forms the Gibbs-Boltzmann distribution (§5.1.3). From (5.35):

$$\mathfrak{G}(\mathbf{x}, T) = \frac{1}{Z(T)} e^{-U(\mathbf{x})/T}$$
$$p_i(x_i) \propto \exp[-V_i(x_i, \mathcal{X}_i)/T]. \quad (5.86)$$

where $Z(T)$ is the partition function. For very large T , the clique potentials $V_i(x_i)$ have comparatively little effect on $p_i(x_i)$, and as $T \rightarrow \infty$, the new state is chosen at chance: $p_i(x_i) \rightarrow 1/n$. Conversely, $T = 0$ is equivalent to a deterministic greedy algorithm, which always selects the state with the largest local probability. An "annealing schedule" dictates how temperature T varies as a function of time t . Large-scale changes may occur much more rapidly for large T because intermediate states with low likelihood may be accepted more easily, and hence T should be large near the start, since this diminishes the influence of the initial state. As T is reduced, the scope for large-scale change diminishes and fewer transitions are expected. The sequence of states \mathfrak{S}_t again forms a Markov chain, with a limiting distribution equal to the desired joint likelihood, reached at $T = 1$. It is possible to set $T = 1$ from the start and dispense with an annealing schedule, simplifying the algorithm marginally, but this increases the influence of the starting state and hence more iterations may be required before transients decay and the desired distribution is reached (Hassner & Sklaasky, 1980). In order to reflect the importance of image "features", a dual region-boundary structure may be adopted (Geman & Geman, 1984).

When the objective is to maximise rather than sample the likelihood, $T = 0$ is appropriate as the destination temperature. Zero temperature during synthesis would always select the state corresponding to the mode of the joint likelihood rather than sampling it fairly. In practice, acceptable synthesis of an image of size 128×128 pixels from a very simple model with no annealing schedule requires about 200 iterations (Geman & Geman, 1984). Fewer iterations are required when the field size is smaller, but a much larger number of iterations is needed to produce realistic images (Derin & Cole, 1986; Derin & Elliott, 1987).

The strengths of the Gibbs sampler lie in its ability to consider many new states simultaneously, both at a single site and across a code. The temperature schedule allows approximate convergence to be reached much more rapidly than would otherwise be the case, and diminishes the influence of the starting state. The Gibbs sampler has

found application outside field synthesis, notably in image restoration and segmentation (§§5.5–5.6), but has heavy computational requirements.

5.3.3 Deterministic Greedy Relaxation

As we have noted (§5.3.1; §5.3.2), stochastic relaxation has the advantage of being able to switch between distant states despite the poor fit of intermediate configurations. A deterministic greedy algorithm allows the fit to be improved *locally* at each time step, and terminates after fewer iterations, but there is a strong danger that the true minimum will not be found and inferior images may result.

A deterministic algorithm was proposed by Galagowicz and Ma (1985), who defined feature vectors corresponding to desired and present image statistics. Each change of state reduces the distance between the feature vectors, and the texture model need not be Markov since the feature vector defines the required attributes. Synthetic copies of natural textures were produced, but the dimension of the feature vector was unfortunately excessive, often comparable with field size, and only 8 gray levels were attempted. In essence, this algorithm is similar to the Gibbs sampler (§5.3.2) at $T = 0$.

5.3.4 Synthesis of Causal Fields

When a Markov random field has a causal neighbourhood, it is straightforward to generate a field from given boundary conditions by application of (5.71). As noted on page 115, a causal form is available for all linear two-dimensional processes although the order of the causal neighbourhood is usually much larger than that of the corresponding non-causal field. Another route is simply to adopt a causal approximation because it is much easier to generate, but this does not generally lead to acceptable results (Hansen & Elliott, 1982).

5.3.5 Synthesis of Auto-Normal Fields

We have hinted above how realisations of homogeneous auto-normal fields may be synthesised (§5.1.5), and this procedure is far quicker than stochastic relaxation methods because the joint distribution may be sampled directly. Under the toroidal boundary approximation, we have shown that the transform of the field $\mathbf{X} \rightleftharpoons \mathbf{x}$ is coloured Gaussian noise (5.69):

$$\mathbf{X}_i \sim \mathbf{G}\left(0, \frac{\sigma^2}{1 - B_i}\right) \quad (5.87)$$

where B_i is the cosine transform of the coefficient vector $\boldsymbol{\beta}$ (5.62). A practical synthesis method is to manufacture the required form from a field of unit-variance white noise \mathbf{v} (Cohen *et al.*, 1991; Kashyap & Chellappa, 1983; Woods, 1972).

$$\mathbf{X}_i = \frac{\sigma \mathbf{v}_i}{\sqrt{1 - B_i}} \quad (5.88)$$

Complex multi-parameter models may be synthesised with this method in only a few seconds of CPU time, often generating images very similar to natural textures (Cohen *et al.*, 1991; Kashyap & Chellappa, 1983). Examples of synthetic Brodatz textures formed by this method are shown in Figure 1.4 (page 7) and Figure 6.2 (page 142). Alternatively, the field may be synthesised as a matrix equation (5.57), or by invoking the causal form (5.71). All these methods are very fast compared with those which do not depend on explicit computation of the joint likelihood.

5.4 Gibbs–Markov Texture Classification

Several approaches to texture classification were discussed in §3.4: test images are usually assumed to contain a homogeneous sample of a single texture type, and the most suitable classification method is influenced by how much information is available from class feature vectors. Since Gibbs–Markov models fully specify the joint distribution, optimal Bayesian classification is feasible (§3.4.1). Given a GMRF model \mathcal{G}_i from class ω_i , it is necessary to compute the joint likelihood of the observed pixel data $\mathcal{P}\{\mathbf{x} | \mathcal{G}_i\} = P_i(\mathbf{x})$. This is straightforward in some special cases, such as an auto-normal GMRF, but is difficult in general unless an approximation is taken (§5.2). If it is impossible to compute $P_i(\mathbf{x})$ directly, some other method should be considered, either approximating the Bayes form or adopting a sub-optimal classification method (§3.4.3). We adopt Bayesian classification as a benchmark to compare our proposed frameworks with a conventional Markovian analysis (Chapter 6), and subsequently suggest designs for modified classifiers to improve robustness with respect to image degradation (Chapter 7).

5.4.1 Modified Bayes Classifiers

It is often desirable to represent each class more flexibly by a parameterised model \mathcal{G}_i , as this may increase classifier robustness or level of abstraction. In this case the missing parameters θ may be estimated from the observed data (Cohen *et al.*, 1991). The modified Bayes classifier substitutes a maximum-likelihood estimate for θ and proceeds as above:

$$\begin{aligned} \mathcal{P}\{\omega_i | \mathbf{x}\} &= \max_{\theta_1, \dots, \theta_n} \mathcal{P}\{\omega_i | \mathbf{x}, \theta\} \\ &= \mathcal{P}\{\omega_i | \mathbf{x}, \hat{\theta}\} \end{aligned} \quad (5.89)$$

where $\hat{\theta}$ is the maximum-likelihood estimate of the missing parameters. Major restrictions on the wider use of GMRF texture models are their scale and rotation variance, but both were partially overcome by a modified Bayesian classifier taking these quantities as free parameters (Cohen *et al.*, 1991). This was feasible because the spectrum of a homogeneous auto-normal model varies predictably with these projection parameters, allowing reliable estimates to be made. High accuracy was reported, but the classifier was only tested with synthetic images which exactly matched the model distribution. Our experience suggests that results obtained with synthetic textures may be unrepeatable with real images (Chapter 6). A more straightforward procedure suffices to protect

the classifier from image-wide changes in brightness, which influence the MRF variance. Pixel and MRF variances are altered by the same proportion and hence their *ratio* is invariant, and should be recorded as a more robust parameter (Kashyap *et al.*, 1982).

5.5 Restoration of Degraded Images

Restoration is concerned with recovering a true copy of a picture that has been degraded, and is a classic image-processing application. We briefly review the use of MRF models for this purpose in order to establish the credentials of Markovian methods for image analysis, and to indicate possible applications for our novel frameworks. Many sources of degradation occur in practice (Rosenfeld & Kak, 1982); we shall make the simplifying assumption that the transformation is linear and hence may be expressed as a matrix equation similar to the form considered above (5-22):

$$\mathbf{g} = \mathbf{H}\mathbf{f} + \mathbf{v} \quad (5-90)$$

where \mathbf{f} is the original field, \mathbf{g} is the observed degraded image, and \mathbf{v} is white noise. When the transform is deterministic and has a known form, the original image may be recovered exactly: $\hat{\mathbf{f}} = \mathbf{H}^{-1}\mathbf{g}$. This is not usually the case, and the algorithm must produce the best estimate for the original image that is available. Estimation of the transform parameters from the degraded image requires assumptions to be made about both the image and transformation.

5.5.1 Maximum *a posteriori* Restoration

Bayesian maximum *a posteriori* (MAP) restoration demands that the conditional likelihood of the true image \mathbf{f} given the degraded copy \mathbf{g} , is maximised.

$$\mathcal{P}\{\mathbf{f} | \mathbf{g}\} \propto \mathcal{P}\{\mathbf{g} | \mathbf{f}\}\mathcal{P}\{\mathbf{f}\} \quad (5-91)$$

The two components of the RHS of (5-91) correspond to the transformation and image likelihoods, respectively, and are generally assumed independent (Rosenfeld & Kak, 1982). Commonly, a particular form is assumed for the true image \mathbf{f} , for example it is often modelled by a Gibbs random field. The transform is parameterised so that it may be evaluated for specific cases, and restoration reduces to maximising the posterior distribution $\mathcal{P}\{\mathbf{f} | \mathbf{g}\}$ with respect to the field \mathbf{f} and transform parameters. This conditional density also follows a Gibbs distribution in practice, and the problem is equivalent to minimising a cost function (Geman & Geman, 1984). It is not always necessary to estimate a parameter set \mathcal{G}_r to describe the true image: an assumption of spatial coherence allows suitable parameters to be manufactured without direct reference to the image (§5.2). With more complicated images, MAP restoration is not always suitable because the search space is enormous and effective maximisation presents a formidable challenge.

Application of the Gibbs Sampler to Image Restoration

Image restoration demands that a good but not necessarily optimal solution should be selected from a vast number of alternatives. As noted in §5.3.2, the Gibbs sampler is well-suited to this type of problem, exploring the vast solution space effectively without explicit computation of the joint likelihood. Exhaustive search is computationally-infeasible because the number of permitted configurations grows explosively with field size and number of gray levels (Gurari & Wechsler, 1982). When it is desired to *maximize* rather than sample the likelihood, the temperature parameter T should be reduced gradually to zero. In practice, T is lowered faster than is consistent with locating the optimal minimum because of computational considerations. Despite this, excellent results have been reported for natural and synthetic images, often at very high noise levels (Geman & Geman, 1984). The main disadvantage of the Gibbs sampler is its requirement for very large amounts of computation, upon which parallel implementation has only limited impact (§5.6.6).

ICM Restoration Algorithm

Besag's (1986) ICM algorithm is a variant of the Gibbs sampler that seeks to reduce the amount of computation at the expense of sub-optimal solutions, and adopts a similar basic approach. Aware that locating the true maximum of the joint posterior density is computationally demanding, Besag proposed iterative maximisation of the *local* conditional densities, exactly equivalent to the Gibbs sampler at $T = 0$. Although successful results were reported (Besag, 1986), and the computational demands are lower than for the full Gibbs sampler, there is no protection against becoming trapped in a poor local-maximum state.

5.5.2 Wiener Filtering

The full complexity of the posterior distribution may be avoided by approximate methods, and Wiener filtering minimises the expected least-squares error between the original and restored image (Rosenfeld & Kak, 1982). If the image joint likelihood is normal, the least-squares function is linear, greatly simplifying the calculations. Assuming the degradation transformation again follows the form (5-90):

$$\hat{\mathbf{f}} = \mathbf{C}_{ff} \mathbf{H}^T (\mathbf{H} \mathbf{C}_{ff} \mathbf{H}^T + \sigma^2 \mathbf{1})^{-1} \mathbf{g} \quad (5-92)$$

where $\hat{\mathbf{f}}$ is the estimate of the original image given the degraded copy \mathbf{g} . Brute-force computation is possible but undesirable because of the dimension of \mathbf{C}_{ff} . If an MRF parameter set is assumed for \mathbf{f} , (5-22) gives:

$$\begin{aligned} \mathbf{C}_{ff} &= \mathbf{H} \mathbf{C}_{gg} \mathbf{H}^T + \sigma^2 \mathbf{1} \\ \hat{\mathbf{f}} &= \mathbf{H}^{-1} (\mathbf{C}_{gg} - \sigma^2 \mathbf{1}) \mathbf{C}_{gg}^{-1} \mathbf{g}. \end{aligned} \quad (5-93)$$

The purpose of this assumption is to allow the covariance matrices to be manipulated easily, and further simplification follows under the toroidal boundary approximation (Chellappa & Kashyap, 1982).

5.6 Segmentation of Textured Images

Unsupervised image segmentation algorithms partition an image into disjoint textured regions, about which little or nothing is known *a priori* (§3.5). This problem is ill-posed in general, and additional information or constraints are required, often in the form of heuristic approximations reflecting the nature of the physical environment (§2.4). In this section, we briefly discuss the use of Gibbs–Markov models to describe image composition, and review some previous approaches to segmentation. Markovian models are capable of representing natural textures sensitively, but drastic approximations are usually taken in a bid to reduce the computational load, detracting from the quality of the resulting segmentations. It is normal, for example, to adopt homogeneous second-order models and pseudo-likelihood manipulation, or even the simultaneous autoregressive form (§3.2.5). Our proposed Gabor–Markov framework and Sampled–Markov models may be used with many of the algorithms described here, but have several important advantages which should strongly reduce the computational burden whilst increasing segmentation quality (§5.7).

5.6.1 Simple Gray-Scale Textures

We shall first consider a class of very simple textures, each taking a constant gray level. Segmentation is non-trivial because of the addition of large amounts of independent random noise, and is virtually identical to restoration in this case. If the noise is white, each texture may be modelled by an MRF of order zero.

Optimal maximum *a posteriori* estimation is appropriate for this type of problem. As above, we shall assume that a “degraded” or textured image \mathbf{g} is available, and the goal is a partition $\hat{\mathbf{f}}$ corresponding to the scene layout (5.90), derived from the posterior density, $\mathcal{P}\{\mathbf{f} \mid \mathbf{g}\}$ (5.91). Maximisation may proceed using the techniques outlined above (§5.5), if “noise” or texture parameters are known (Hansen & Elliott, 1982). The arrangement of textured regions is not usually known accurately, but may be described adequately by a clustering GRF. Without this influence of spatial context, a per-pixel MAP estimate would give very noisy results which would require smoothing or some other form of post-processing (§3.5.3).

In general, noise parameters are unknown, and may be inhomogeneous. All the estimation methods discussed above (§5.2) require access to a contiguous region formed from a single texture type, but this is not generally available until the segmentation has been completed. In order to overcome this, a more complex image model is required.

5.6.2 Hierarchical Random Field Representations

A monolithic model is sufficient to capture the structure of the heavily-simplified textures described above (§5.6.1), when the complete image may be viewed as being composed of a single meta-texture. In effect, we have described a random mosaic model, each texture primitive having constant intensity (§3.3.3). Image structure is generally much more complex and may take different forms at different levels of description (Marr, 1982), for which a hierarchical model is required. We shall limit ourselves to two levels of

structure, corresponding to a model for each individual texture and for the arrangement in which they appear in the scene (§3.1). An example of this type of decomposition was shown in Figure 4.13 (page 77). We have shown above that Gibbs–Markov random fields provide suitable models for single textures; this extension allows their use for more realistic images. It is not necessary for models at both levels to be Gibbs–Markov but this is convenient and is often done.

Given an observed image \mathbf{y} , the goal is to compute an estimate of the scene partition, \mathbf{x} . Of course, \mathbf{x} cannot be observed directly, but a MAP estimate is obtained by maximising the posterior density:

$$L(\mathbf{x} | \mathbf{y}) = L(\mathbf{x}) + L(\mathbf{y} | \mathbf{x}) \quad (5-94)$$

dropping constant terms. This maximisation problem is very similar to that encountered during image restoration, with the added difficulty that model parameters for each texture type are unknown. Use of a scene model to bias segmentations towards “reasonable” partitions gives superior results to data-independent techniques such as median filtering, although the latter is much faster (Hansen & Elliott, 1982).

5.6.3 Non-Adaptive Segmentation Algorithms

Parameters for both scene and texture models are known or assumed in advance by non-adaptive segmentation algorithms, or estimated independently. The crux of these methods lies in the computational problem of minimising the posterior likelihood (5-94), for which exhaustive search is computationally infeasible (§3.5).

Two-Stage Region Splitting Algorithm

An ingenious two-stage algorithm was proposed by Cohen and Cooper (1987), who used second-order auto-normal texture models and an auto-binary scene model. Their approach is of the region-splitting variety described above (§3.5), and the image is initially decomposed over a pyramidal grid following a quad-tree pattern. Each block is initially assumed to contain a single texture type and is classified accordingly against known parameter sets (§5.4), at successively finer resolution. Errors are most likely near the start of the process when the blocks are large and are unlikely to contain a single homogeneous texture. Some pixel values from adjacent blocks are required in order to complete the neighbourhood sets of pixels near the border, and a difficulty arises when the required sites are occupied by a different texture type because the likelihood function then assumes a different form. This is overcome by temporarily replacing such pixels by the field mean, and proceeding as before. Termination follows when the size of the blocks approaches pixel dimensions. Noting that most of the difficulty in computing the likelihood (5-55) lies in the determinant, Cohen and Cooper (1987) used the toroidal form (5-64) for this term whilst retaining the true neighbourhood structure for the remainder of the expression.

The segmentation produced by the first stage is coarse with jaggy boundaries created as artifacts of the quad-tree structure, and no account is taken of the scene model in its formation. A deterministic greedy restoration algorithm refines it during the second

stage (§3.5.3), except that the state is perturbed and ascent re-started on convergence, as a safeguard against poor local maxima.

Dynamic Programming Algorithm

An alternative approach to an essentially computational problem employs dynamic programming (Derin & Elliott, 1987). Pixel interaction is assumed to be predominantly local, and correlation between pixels separated in the image by more than a certain distance is assumed negligible. Exploiting this, the optimal scene configuration within a narrow strip may be calculated exhaustively, conditioned on the previously-segmented bordering half-plane. The line of pixels adjacent to this plane is classified according to this local maximum state, and the algorithm advances by one row. The image is processed semi-sequentially, discarding intermediate results at each stage as the context is enlarged. Even when the width of the strip and the number of texture types are small, this algorithm is computationally-demanding. A second-order GRF was used for both texture and scene models.

A mild relaxation of the condition that all image parameters must be given *a priori* was offered by Lakshmanan and Derin (1989): texture parameters are still fixed, but the scene layout parameters may be adapted. The algorithm initially proceeds as for the non-adaptive version, but scene parameters are periodically re-estimated from the current segmentation, and the relaxation re-started, eventually terminating after a fixed number of iterations. Although high accuracy was reported with some artificial textures (Lakshmanan & Derin, 1989), the algorithm is computationally intense and the degree of adaptability is very small.

5.6.4 Adaptive Segmentation Algorithms

Each of the above algorithms demands that all texture parameters are known in advance and that no degradation occurs, but both conditions are unreasonable in practice. A clustering algorithm which overcomes these restrictions was proposed by Silverman and Cooper (1988), who initially divide the image into small blocks (§3.5.1). Each is assumed to contain a single texture type, for which a second-order auto-normal pseudo-likelihood parameter set is estimated. Using a Mahalanobis distance criterion (§3.4.2), adjacent blocks with compatible parameter sets are merged using a greedy algorithm, building up a coarse segmentation of the image. Common manufactured objects have smooth surfaces, and these were described more closely by adopting a non-stationary Markov model, whose field mean varied according to a low-order polynomial. Estimation of the polynomial coefficients complicates extraction of other MRF parameters as the two processes are not independent. The coarse nature of the segmentation resulting from agglomerative clustering means that a second pass to “restore” it is desirable (§3.5.3), but even when this is done, real and artificial scenes may be segmented without excessive amounts of computation (Manjunath & Chellappa, 1991).

5.6.5 Multiple-Resolution Segmentation Algorithms

All the segmentation algorithms described above suffer from a requirement for large amounts of computation, despite approximations taken to reduce this load. Multiple-resolution algorithms address this weakness by obtaining initial coarse estimates across an image block and subsequently refining them at the pixel level (§3.5). Multi-grid algorithms extend this approach by conducting the relaxation at several scales simultaneously (Terzopoulos, 1986), and this was the basis of the multiple-resolution adaptive segmentation algorithm proposed by Bouman and Liu (1991). An initial coarse segmentation is obtained after decomposition of the image into small blocks as above (§5.6.4), and the resulting scene is used as the initial state for a greedy reconstruction algorithm which is applied successively at each resolution. After each change, parameters are re-estimated and the algorithm re-started. Segmentation accuracy is intermediate between the Gibbs sampler and ICM algorithm (§5.5.1), but less computation is required than either, representing an acceptable compromise.

5.6.6 Approaches to Parallel Implementation

A common feature of many algorithms described in this Chapter is the requirement for large amounts of computation, sometimes leading to excessively long execution times on conventional sequential processors. Parallel implementation appears to present an attractive means of addressing this concern (§A.1), but *exact* solution remains infeasible (Gurari & Wechsler, 1982; Tsotsos, 1987), and we consider parallel approaches to efficient sub-optimal algorithms. A classic example is the Gibbs sampler (§5.3.2), claimed by its authors to be parallel although their implementation was sequential (Geman & Geman, 1984). The state of each pixel is iteratively updated by a relaxation process according to the value taken by other pixels within its Markov neighbourhood (§5.1.2), and consequently image codes must be processed sequentially but pixels within each code may be updated in parallel (§5.2.2). During image restoration, processing requirements are local, regular, and possess translational symmetry; and hence may be met efficiently by data-parallel architectures (Derin & Won, 1987; Murray *et al.*, 1986).

Image segmentation algorithms are necessarily less homogeneous, and hence map less naturally onto data-parallel hardware (§A.1.1). This is particularly true for adaptive algorithms, which combine parameter estimation from irregular regions with boundary localisation, and their more loosely-constrained processing requirements are supported flexibly by task-parallel hardware (§A.1.2). Both these approaches exploit spatial parallelism to some degree, but are partially sequential in one sense because very many iterations are usually required at each processor.

5.7 Motivation for Gabor–Markov Framework

Natural textures often exhibit random spatial variation, and the stochastic influence of local image context is made explicit by Markovian representations which describe each pixel by a conditional probability distribution. Algorithms manipulating Gibbs–Markov models generate competing statistical hypotheses about image composition or layout,

which may be formally compared by evaluating their respective joint likelihoods. This is computationally burdensome in general, but is more straightforward for homogeneous auto-normal models which are frequently adopted for this reason, often in conjunction with the toroidal boundary approximation. Explicit calculation of the joint likelihood is not imperative, but stochastic algorithms which manipulate only the conditional likelihood are often very computationally demanding.

The descriptive power of Gibbs–Markov models was established by reviewing some previous applications in the literature, often involving the manipulation of ill-constrained natural textures. Computational concerns frequently motivated the use of low-order models, able to describe only very local spatial correlation. Sub-optimal pseudo-likelihood parameter sets were frequently adopted for the same reason, detracting from the “optimal” statistical framework. Hierarchical random field models describe the image more richly by representing structure at several levels, but are even more computationally exacting. Despite these concerns, Gibbs–Markov models have been successfully employed in a variety of demanding applications, including image restoration and unsupervised segmentation.

5.7.1 Drawbacks of Gibbs–Markov Models

Despite the apparent success of Gibbs–Markov models in the applications reviewed above, they suffer from a number of limitations which restrict their wider use for texture analysis.

1. A major disadvantage of Gibbs–Markov models is their computational appetite. Even approximate solutions obtained with low-order models and pseudo-likelihood parameter sets cannot be achieved in real time. This is in contrast to spatially-parallel algorithms, which may be executed extremely efficiently by suitable architectures (§3.2).
2. Raw Gibbs–Markov models perform stochastic template-matching on the observed image data, and hence alterations of viewpoint, lighting, texture mapping, or the image acquisition environment all require appropriate adjustment of model parameters. While it is often desirable to detect these changes, it is seldom useful for them to create distinct textures.
3. None of the papers reviewed in this Chapter attempted to assess the robustness of Gibbs–Markov models to image degradation by blur or noise: all apart from the simple gray-scale textures (§5.6.1) were assumed to be noise-free, but this is hardly realistic in practice. Our experiments show that raw Gibbs–Markov models are very sensitive to these artifacts, particularly when using pseudo-likelihood parameter sets (Chapter 6).
4. Multi-resolution algorithms seek to arrive quickly at a coarse solution and then refine it to achieve higher accuracy. Despite recent developments (§5.6.5), this strategy has not been fully exploited and most algorithms operate solely at the pixel level.

5.7.2 Proposed Improvements: Gabor–Markov Framework

Our objective is to retain the descriptive power offered by conventional Markovian analysis, but to raise computational efficiency, robustness, and level of abstraction (Chapter 1). Motivated by the apparent success of simple local image “features” in pre-attentive human vision (Chapter 2), and the computational efficiency of “statistical” approaches (Chapter 3), we assessed experimentally the potential of Gabor filters for image analysis (Chapter 4). Suitably constrained images were segmented accurately by locating first-order differences in Gabor amplitude, but this approach is unreliable because considerable variation is often observed within as well as between textured regions. Rather than attempting to suppress this by heuristic post-processing, our approach is to describe the spatial variation of Gabor features with Markov random fields, forming a hybrid Gabor–Markov framework for texture analysis. We anticipate that this dual paradigm analysis will allow us to combine the best aspects of “statistical” and “structural” approaches in a similar manner to the pre-attentive dichotomy observed in low-level human vision. In particular, Gabor–Markov models address the concerns noted above (§5.7.1).

1. Feature values have a higher information content than pixels, allowing the feature array to be sub-sampled without significant loss of information. A reduction in the effective image size immediately leads to computational benefits, allowing execution times to be reduced or models of a higher order to be used.
2. Our Profile and Resultant feature-extraction algorithms generate sensitive and compact descriptions of real textures, but require only simple data-parallel processing and hence may be implemented very efficiently on suitable architectures. Straightforward enhancements would allow them to adapt to observed image characteristics, providing a welcome degree of abstraction so that feature representations always assume a standard form.
3. Each feature vector has a region of support much larger than a single pixel and hence should be less affected by image degradation, suggesting that a feature-based model should be more robust.
4. Gabor filtering may be used to derive an approximate image segmentation for low computational cost (§4.4), which can be used to guide selectively a more expensive Markovian analysis.

We propose specific Gabor–Markov models in Chapter 6, and compare their performance with a conventional Markovian analysis by adopting Bayesian classification as a benchmark.

6

Markovian Texture Classification: A Comparison of Novel and Conventional Models

Gabor filtering is a powerful tool for texture analysis. It is a promising member of the class of texture energy measures, which may be computed in a data-parallel fashion with a local region of support (§3.2). In Chapter 4, we described the implementation of a simple spatially-parallel image segmentation algorithm based directly on Gabor filter amplitude. Performance was adequate for simple images, but a number of deficiencies that limited its scope were noted. Only first-order statistics of Gabor amplitude were used by this method — spatial context was neglected. In practice, Gabor filtering often failed to eliminate variation within textured regions, and these residual fluctuations complicated the process of boundary extraction. We developed models for Gabor filter output in terms of response signatures of simple image primitives, and proposed two efficient feature-extraction algorithms.

One factor which limits the applicability of simple spatial-filtering algorithms for texture analysis is their failure to acknowledge an inherent characteristic of real textures: variability. This attribute is represented explicitly by Markov¹ random field (MRF) models. Each pixel is described by a conditional probability distribution, expressing the stochastic influence of spatial context (Chapter 5). Markov models have been employed successfully in a variety of demanding applications, including image restoration and unsupervised segmentation, and their descriptive power is well-established. Their main

¹Whilst acknowledging the Gibbs-Markov equivalence (§5.1.3), we shall only make use of the Markov formulation in this Chapter.

drawbacks arise from heavy computational requirements, which cannot be met fully by parallel implementation, and an insufficient level of abstraction.

In this Chapter, we combine Gabor filters with Markov fields to form a new method for texture analysis. Arrays of Gabor features obtained by our extraction algorithms are represented by Markov random fields to form the new class of *Gabor-Markov* models. This hierarchical arrangement should allow us to exploit the spatial variability of Gabor amplitude noted above. Different modelling assumptions lead to the implementation of several members of this hybrid framework. We hope to retain some of the speed and flexibility of raw Gabor filtering coupled with the superior scope and performance offered by Markov models. The efficacy of this approach is measured by comparing the performance of our proposed Gabor-Markov method with a conventional Markovian analysis. Texture classification accuracy was chosen as a suitable benchmark. Despite some criticism of its generality (§3.6.1), empirical classification accuracy gives a simple quantitative measure of performance, may easily be repeated for different parameters, and does not require excessive amounts of computation.

Relative classification accuracy of the Gabor-Markov and conventional MRF models is assessed using 33 textures chosen from the popular Brodatz album, divided into 132 images. Performance gains anticipated for our new Gabor-Markov method are observed in practice: image dimensions may be reduced by a factor 16 without loss of accuracy; noise tolerance is improved by a factor of up to 45; blur tolerance is increased considerably (§6.5).

Our appraisal of Gabor-Markov models leads to a greater appreciation of the importance of suitable pre-processing. We propose a *Sampled-Markov* framework, employing concise representations derived directly from the image data (§6.6). Performance with this paradigm further improves upon a conventional Markovian analysis. Image dimensions may be reduced by a factor 25 without loss of accuracy, and noise tolerance is improved by a factor of up to 200. Smooth-Sampled Markov models are preferred, and achieve 100% accuracy in our tests.

These results are of considerable practical interest. The superior performance of Sampled-Markov classifiers is particularly remarkable. We investigate possible causes for these trends in Chapter 7. Finally in Chapter 8, we propose modifications which should further enhance the performance of both Sampled- and Gabor-Markov classifiers, and discuss potential applications.

6.1 Choice of Auto-Normal MRF Model

From the results given in the previous Chapter, it is clear that auto-normal Markov random fields are particularly convenient to manipulate in practice, compared to other forms of MRF. It is feasible to evaluate the joint likelihood; the conditional probability distribution assumes a simple form, with a linear neighbourhood function; and analysis is often possible using analytical rather than stochastic numerical techniques. Homogeneous auto-normal models are therefore very much favoured candidates for image description, and their benefits are so overwhelming that we have confined our attention

exclusively to this form. Whether this choice is appropriate depends on the balance between computational tractability and the descriptive power of the model. Assumptions that images are multivariate normal are common (Bouman & Liu, 1991; Chen & Pavlidis, 1983), and have some empirical support (Hunt & Cannon, 1976).

In the context of a pixel image, a number of approximations are inherent in the choice of an auto-normal MRF: the normal distribution is continuous and unbounded whereas the image is discrete and bounded, taking integer gray-levels in the range 0-255, and some further inconsistencies are described below (§7.1.1). Despite this, auto-normal models are popular in the literature and have been shown to be effective practical tools, if the approximations are not taken to extremes (Chellappa & Chatterjee, 1985; Cohen & Cooper, 1987).

We shall also take the approximation of toroidal boundary conditions (§5.1.5). Each pixel possesses a surrounding neighbourhood set, whose values are required to form its conditional gray-level distribution, but some of these values are missing when the centre pixel lies near an image boundary. Dimensions of the images considered below are very large compared to those of the neighbourhood set, and hence the toroidal boundary approximation introduces negligible error. We chose it because it is more convenient than the free boundary condition, particularly when using discrete Fourier transforms. Since our images are all rectangular, no difficulty arises with an irregular boundary shape. We further assume that fields are stationary: their mean does not vary as a function of lattice site.

6.1.1 Other Possible MRF Models

Experiments with other distribution functions have been limited in the literature because of the attractiveness of auto-normal models. The binomial distribution is discrete and bounded, and has been employed to model pixel images (Cross & Jain, 1983), but computational difficulties forced the use of only 8 gray levels rather than the usual 256. Other discrete models also suffer from this restriction (Gagalowicz & Ma, 1985), which drastically reduces their descriptive power.

6.2 Pixel Images: Conventional Analysis

The main set of images used with the classifier was digitised from the Brodatz album. Camera and lighting parameters were usually adjusted to ensure that the dynamic range was well used but some images have been re-normalised. No corrections were made for camera transfer characteristics. Unless stated to the contrary, the original images were 256×256 pixels (Figure 1.6 on page 11).

6.2.1 Estimation of MRF Parameters

As discussed in §5.2, several parameter estimation methods are available. Two were used here: maximisation of the true $P(\mathbf{x})$ and pseudo $\tilde{P}(\mathbf{x})$ likelihoods.

$$\begin{aligned}
 P(\mathbf{x}) &= \sqrt{\frac{|\mathbf{B}|}{(2\pi\sigma^2)^{MN}}} \exp\left[-\frac{(\mathbf{x} - \boldsymbol{\mu})^T \mathbf{B}(\mathbf{x} - \boldsymbol{\mu})}{2\sigma^2}\right] \\
 \tilde{P}(\mathbf{x}) &= \left(\frac{1}{2\pi\sigma^2}\right)^{MN/2} \exp\left[-\frac{1}{2\sigma^2} \sum_{i \in \mathcal{L}} \left(x_i - \mu - \boldsymbol{\beta}^T(\boldsymbol{\eta}_i - 2\boldsymbol{\mu})\right)^2\right] \quad (6.1)
 \end{aligned}$$

Our notation is as used previously (Chapter 5): \mathbf{x} image data defined on lattice \mathcal{L} , ordered as an $M \times N$ vector; $\boldsymbol{\mu} = \mu \mathbf{1}$ mean vector; $\boldsymbol{\beta}$ MRF neighbourhood coefficients, ordered as a column vector; \mathbf{B} a block-circulant matrix formed by rotating $\boldsymbol{\beta}$; $\boldsymbol{\eta}_i$ the neighbourhood vector for point i , defined by (5.51) on page 112; P true image likelihood; \tilde{P} image pseudo-likelihood; σ^2 MRF predictor variance; $\mathcal{S} = \{\mu, \sigma^2, \boldsymbol{\beta}\}$ MRF parameter set. Maximum-likelihood parameter estimates are obtained by maximising the appropriate likelihood function (6.1) with respect to the free parameters, \mathcal{S} (§5.2.1). In order to improve clarity, the mean will be dropped from now on. The corresponding log-likelihood functions $L(\mathbf{x})$ and $\tilde{L}(\mathbf{x})$ are more convenient theoretically and practically.

When using the Group method (§6.4.3), simultaneous parameter estimation from several images is required. Taking these images to be independent, the new joint likelihood may be written as the product of those for each image.

$$\begin{aligned}
 P(\mathbf{x}^{(0)}, \mathbf{x}^{(1)}, \dots, \mathbf{x}^{(n-1)}) &= \prod_{i=0}^{n-1} P(\mathbf{x}^{(i)}) \\
 L(\mathbf{x}^{(0)}, \mathbf{x}^{(1)}, \dots, \mathbf{x}^{(n-1)}) &= \sum_{i=0}^{n-1} L(\mathbf{x}^{(i)}) \quad (6.2)
 \end{aligned}$$

The notation $\mathbf{x}^{(i)}$ refers to the i -th image.

Pseudo-Likelihood Parameter Estimation

For an auto-normal MRF, the pseudo-likelihood parameter estimate is the same as the least-squares estimate, and is relatively straightforward to evaluate (§5.2.5). Group estimates for n $M \times N$ images are:

$$\begin{aligned}
 \tilde{\mu} &= \frac{1}{nMN} \sum_{j=0}^{n-1} \sum_{i \in \mathcal{L}} x_i^{(j)} \\
 \tilde{\boldsymbol{\beta}} &= \left[\sum_{j=0}^{n-1} \sum_{i \in \mathcal{L}} \hat{\boldsymbol{\eta}}_i^{(j)} \hat{\boldsymbol{\eta}}_i^{(j)T} \right]^{-1} \sum_{j=0}^{n-1} \sum_{i \in \mathcal{L}} \hat{\boldsymbol{\eta}}_i^{(j)} \hat{x}_i^{(j)} \\
 \tilde{\sigma}^2 &= \frac{1}{nMN-1} \sum_{j=0}^{n-1} \sum_{i \in \mathcal{L}} \left(\hat{x}_i^{(j)} - \boldsymbol{\beta}^T \hat{\boldsymbol{\eta}}_i^{(j)} \right)^2 \quad (6.3)
 \end{aligned}$$

where the notation $\hat{\mathbf{x}}$ refers to a centered process, formed by subtracting the appropriate mean from \mathbf{x} . Unless the context requires the use of this notation, we will otherwise

assume that the field has been centered. The largest neighbourhood size used was ninth order, which required the inversion of a 24×24 matrix. This presents no practical difficulties.

Although the pseudo-likelihood estimate is an approximation, it is often able to capture much of the character of an image at relatively modest computational cost. One significant disadvantage is that the parameter estimate need not correspond to a homogeneous texture ($\hat{\mathbf{B}}$ not positive definite), and often the homogeneity constraint (5-66 on page 115) does not hold. Whenever homogeneity is assumed (as in the synthesis procedure described below, §6.2.2), a contradiction occurs.

True-Likelihood Parameter Estimation

True maximum-likelihood parameter estimates are superior to pseudo-likelihood estimates but are generally available only at considerably increased computational cost. Unfortunately, no analytic expression for the optimal parameter vector is available, and the likelihood must be maximised mainly by numerical techniques (page 119).²

$$L(\mathbf{x}^{(0)}, \dots, \mathbf{x}^{(n-1)}) = \frac{n}{2} \log |\mathbf{B}| - \frac{nMN}{2} \log(2\pi\sigma^2) - \frac{1}{2\sigma^2} \sum_{i=0}^{n-1} \mathbf{x}^{(i)\top} \mathbf{B} \mathbf{x}^{(i)}$$

$$\hat{\sigma}^2 = \frac{1}{nMN-1} \sum_{i=0}^{n-1} \mathbf{x}^{(i)\top} \mathbf{B} \mathbf{x}^{(i)} \quad (6-4)$$

Substitute for $\hat{\sigma}^2$.

$$L(\mathbf{x}^{(0)}, \dots, \mathbf{x}^{(n-1)}) = \frac{n}{2} \left[\log |\mathbf{B}| - MN \log \left(\frac{2\pi \sum_{i=0}^{n-1} \mathbf{x}^{(i)\top} \mathbf{B} \mathbf{x}^{(i)}}{nMN-1} \right) - \frac{nMN-1}{n} \right] \quad (6-5)$$

Under toroidal boundary conditions, the coefficient matrix \mathbf{B} is block-circulant, and hence (6-5) may be simplified considerably. Dropping constant terms, the expression to be maximised is:

$$L(\mathbf{x}^{(0)}, \dots, \mathbf{x}^{(n-1)}) = \sum_{i \in \mathcal{C}} \log \lambda_i - MN \log \left[\sum_{i \in \mathcal{C}} \lambda_i \left(\sum_{j=0}^{n-1} |X_i^{(j)}|^2 \right) \right] \quad (6-6)$$

where (5-61)

$$\lambda_{m+nM} = 1 - 2 \sum_{(k,\ell) \in \mathcal{N}^+} \beta_{k+tS} \cos \left[2\pi \left(\frac{mk}{M} + \frac{n\ell}{N} \right) \right].$$

Summation for i extends over the image lattice \mathcal{L} ; S represents the span of the MRF neighbourhood \mathcal{N}^+ . Power spectra and cosine terms may be pre-computed, and summation over the image may be reduced by half by taking advantage of conjugate symmetry.

²One degree of freedom is subtracted from the expression for the estimated variance $\hat{\sigma}^2$ because the field mean is also determined empirically (Papoulis, 1990, page 222). This ensures that the estimator is consistent: $\mathcal{E}\{\hat{\sigma}^2\} = \sigma^2$.

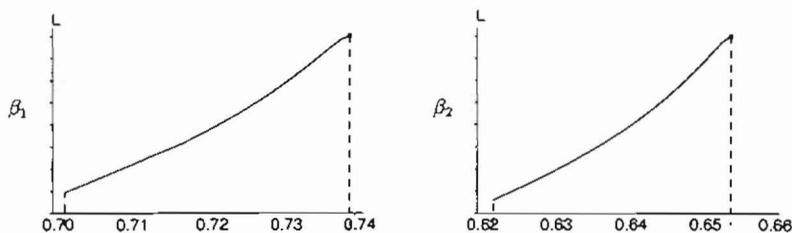


FIGURE 6.1: **Shape of the true-likelihood function near the maximum.** A ninth-order model was estimated from the pressed cork image (#4; Figure 1.6 on page 11). The effect on the log-likelihood L of perturbing the first two elements of $\hat{\beta}$ around their estimated values is shown. Coefficients mesh together so that the scope for independent variation is small, because of the requirement to observe the homogeneity constraint (5-66; page 115). When this is broken, the likelihood becomes zero ($L \rightarrow -\infty$), shown by dotted lines.

Partial gradients of (6-6) with respect to β are easily computed and may be used in a Newton gradient-ascent algorithm. We have found this process to yield satisfactory parameter estimates.

Two problems remain: the first is that the gradient-ascent algorithm sometimes attempts to evaluate (6-6) at a point where the homogeneity constraint $\lambda_i > 0$ (5-66 on page 115) does not hold. Since this parameter set is inadmissible, one potential action is to trap the numerical exception and to set $L = -\infty$. Unfortunately, no gradient can then be computed, and the algorithm fails. A second problem is that the algorithm will probably converge to a *local* maximum, and there is no practical method for comparing this with the true maximum. Stochastic methods may be used to perform a thorough probabilistic search, but this is very slow (Geman & Geman, 1984; Kirkpatrick *et al.*, 1983).

Both these difficulties are overcome by the use of an appropriate seed point. Since the pseudo-likelihood estimate $\hat{\beta}$ is acceptable and fairly cheap to compute, this is used to generate an initial estimate $\hat{\beta}_0$. When necessary, $\hat{\beta}$ is scaled uniformly to ensure that $\hat{\beta}_0$ satisfies the homogeneity constraint (5-66). Under these conditions, we have found that the Newton algorithm converges to an acceptable estimate, usually taking no more than two minutes per image, sometimes much less.³

The parameter estimation algorithm was checked in several ways. Firstly, parameters re-estimated from synthesised images were always very close to the original values. Secondly, when the algorithm was restarted from a random initial point ten times, in no case was a superior maximum found. As final confirmation, the likelihood function (6-6) was plotted as a function of parameter values (Figure 6.1). Each was perturbed around its "maximum" value to assess how local this extremum was. Unfortunately, this proved to be a weak test because the homogeneity constraint (5-66) imposes a very tight limit

³This figure is for estimation of ninth-order parameters from an image of size 128x128 pixels, running on a Sun SPARC architecture. For comparison, pseudo-likelihood estimation takes only 15 seconds.

on the values which each coefficient may assume, such that the scope for individual variation is very small.

6.2.2 Texture Synthesis from Measured Parameter Sets

As a final qualitative check, images were synthesised with the estimated parameters and compared visually with the originals. The Fourier transform synthesis method was used (§5.3.5), again taking advantage of toroidal boundary conditions. In order to allow the approximate visualisation of pseudo-likelihood parameter estimates which do not obey the homogeneity constraint (5.66), the spectral-density coefficients λ_i were forced to be non-negative.

$$\lambda_{m+nM} = \left| 1 - 2 \sum_{(k,\ell) \in \mathcal{M}^+} \beta_{k+\ell S} \cos \left[2\pi \left(\frac{mk}{M} + \frac{n\ell}{N} \right) \right] \right| \quad (6.7)$$

Although this alters the appearance of the image, the effect is usually small when only a few terms are affected. Unfortunately, the transform of the synthetic image is proportional to $\sqrt{\lambda_i^{-1}}$ and hence is very sensitive to small perturbations around $\lambda_i \approx 0$. Each synthesised image is quantised and its range restricted to $[0, 255]$ by clipping at the extremes.

Examples of some ninth-order synthetic Brodatz textures are shown in Figure 6.2; the originals are in Figure 1.6 on page 11. In all cases, no valid point-to-point comparisons can be made between original and synthetic textures. When the real texture appears homogeneous, the true-likelihood synthetic texture is often a very good likeness (e.g. #4, #24, #38, #70). Similar trends were noted in the fifth-order synthetic Brodatz textures shown earlier (Figure 1.4 on page 7). If the original possesses significant structure extending over a region larger than the MRF neighbourhood, however, this is not reflected in the synthetic texture, which is instead dominated by microtexture (e.g. #15, #17, #65, #95). In these cases, it is not surprising that the model has failed to capture the inhomogeneity of the real texture. This failure is only partial, however, as the microtexture may still provide sufficient information for accurate discrimination.

Banding in some synthetic pseudo-likelihood textures is caused by the partial recovery by the synthesis algorithm from an inhomogeneous model. The spectral-density coefficient λ_i has become very small, and possibly negative, at the offending frequencies, causing these to dominate the synthetic image. An alternative (but more costly) synthesis procedure could reduce or eliminate these artifacts. Our main goal is classification rather than synthesis, however, during which a different method (page 147) is used to overcome violations of the homogeneity constraint.

6.3 Feature Images: Gabor–Markov Models

Texture analysis is often divided into statistical and structural approaches: the former represent a texture indirectly by a derived property; the latter by the spatial arrangement of primitive elements. From our review of current texture methods in Chapter 3,

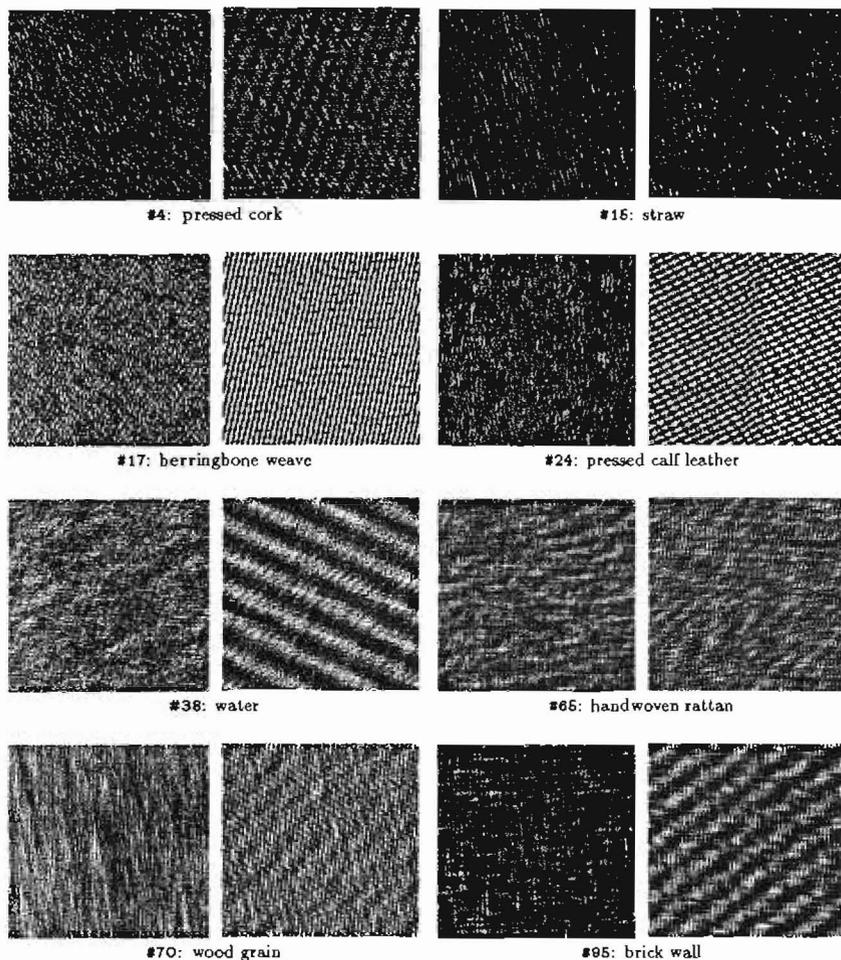


FIGURE 6.2: **Synthetic Brodatz images.** Ninth-order copies of natural Brodatz textures, 256×256 pixels, formed by the MRF method described in §6.2.2: *left*—true likelihood; *right*—pseudo likelihood. No point-to-point comparison with the originals in Figure 1.6 on page 11 is possible. The auto-normal MRF model usually characterises the texture well unless its structure extends over a region greater than the MRF neighbourhood. Textures formed with true-likelihood parameters are consistently more convincing.

we concluded that statistical approaches have the advantage of computational simplicity whilst structural techniques are more sensitive to the hierarchical nature of natural textures, but often require large amounts of computation. A hybrid stochastic–structural model was proposed to combine the attractive properties of both approaches. Gabor filtering and Markov random fields were identified as particularly promising techniques, and were examined in detail in Chapters 4 and 5, respectively. We are now able to propose how our hybrid model should be implemented.

Our objective is to represent arrays of Gabor features extracted by the filtering methods proposed in Chapter 4, by Markov random fields, forming *Gabor–Markov* models. As with pixel data, using an auto-normal MRF is almost obligatory on computational grounds. Features convey more information than individual pixel values, and we anticipate that computational benefits will arise from sparse sampling (§5.7). Conventional Markov models operate at a very low level, and may suffer from a lack of abstraction. Describing the spatial arrangement of features rather than pixels addresses this concern, and may bring similar advantages to those claimed for generalised co-occurrence matrices (§3.3.1; Davis *et al.*, 1981). In particular, aspects of performance measured in this Chapter are: computational efficiency, determined by the degree of sampling; and robustness to image corruption by noise and blur.

Several features are produced by both Profile and Resultant extraction algorithms (§4.6). One possibility is to assume features are independent.

$$P(\theta, c, \epsilon) = P_\theta(\theta)P_c(c)P_\epsilon(\epsilon) \quad (6.8)$$

In practice, the error field ϵ is not used. Independence has the advantage that the contribution for each term may be analysed in isolation, and features with little diagnostic information may be dropped. For example, if the orientation field θ were found to be the most reliable discriminant, we could use $P(\theta)$ alone. This approximation also facilitates modelling with independent MRFs. Feature arrays are not truly independent, however, and this approach is not always appropriate.

6.3.1 Angular Field Models

Our goal is to model an angular pattern θ by means of an auto-normal MRF. The Gabor orientation feature obtained according to §4.6 is continuous but its range is restricted to $0 \leq \theta < \pi$. This may easily be extended to cover any segment of size 2π by a simple transform, and we shall assume that this has been done for the remainder of this section (6.3).

Direct Normal Model

As with the pixel images, the simplest use of the auto-normal MRF is to model the image data directly, but the fit to an angular pattern is unfortunately very poor. Figure 6.3 shows representations of the original and ninth-order true-likelihood reconstructed angular components of the Resultant orientation field extracted from the pressed cork image (#4; Figure 1.6 on page 11). It is not surprising that the fit is poor, because the normal model ignores the periodic nature of the data: whereas $-\pi$ and π^- should be

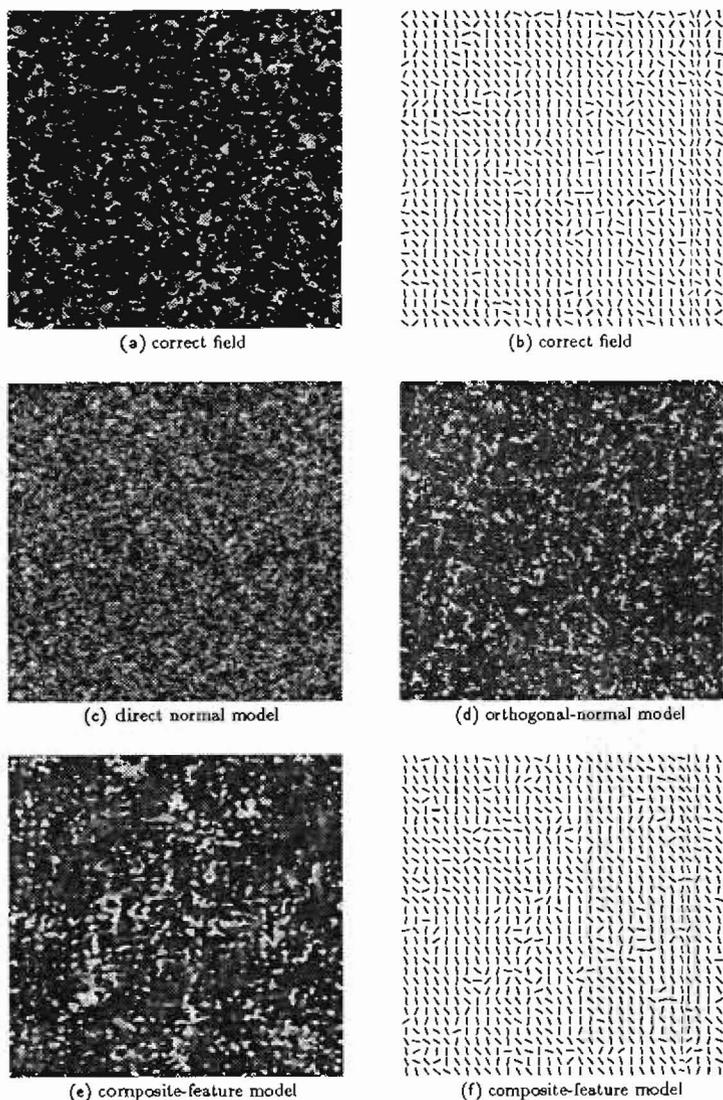


FIGURE 6.3: Comparison of Gabor-Markov angular field models. Gabor features for the pressed cork image (#4; Figure 1.6 on page 11) were extracted by the methods described in §4.6. White represents $\theta = 0$ up to black for $\theta = \pi$. (a) unsampled orientation field. (b) needle diagram for (a). (c-e) true-likelihood ninth-order synthetic copies of (a). (f) needle diagram for (c). The orientation field synthesised according to the composite-feature model is most similar to the original.

adjacent points, they are instead treated as extremes. This distorts the structure of the pattern, and means that the direct auto-normal approximation is of no practical use. The synthetic angular field shown in Figure 6.3c is displayed using the same gray-scale as for the original. Values exceeding the normal pixel range were clipped at the limit rather than being wrapped around.

A possible alternative is to apply a periodic transformation:

$$\tau = \tan(\theta/2). \quad (6-9)$$

This solves the range problem but $-\pi$ and π^- are still treated as poles.

Orthogonal Normal Fields Model

Most of the advantages of the auto-normal distribution may be retained whilst acknowledging the periodicity of the data if orientation is represented by the direction of a vector. The resolute of the vector along the principal axes may then be modelled as normal. Since only the direction of the vector has any significance, there are no restrictions on the range of its components.

Let the two orthogonal components be p and q . We then have:

$$\theta = \arctan(p/q) \quad (6-10)$$

for $-\pi < \theta \leq \pi$. Effectively (6-10) is a partial transformation to polar co-ordinates. The marginal distribution $f_{\theta}(\theta)$ is derived in §A.4, and a graph of this function for the case $\sigma_p = \sigma_q$ is shown in Figure A.7 (page 268). This appears to have greater potential for describing the angular distribution than the direct normal model although this must be verified empirically.

Pseudo-inversion of (6-10) for a given θ gives:

$$\begin{aligned} p &= \sin \theta \\ q &= \cos \theta \end{aligned} \quad (6-11)$$

implying that $-1 \leq p, q \leq 1$ and $p^2 + q^2 = 1$. This relation (6-11) is used to form the training data p and q , but is incompatible with them being normal and independent. Ambiguity about the scaling for p and q occurs since only their ratio is specified by (6-10).

Because the fields p and q are assumed normal and independent, their MRF parameters may be estimated from the training data given by (6-11) using the method of §6.2.1. As they are estimated in isolation, the two parameter sets have freedom to be completely different, but are similar in practice. Following the same argument, an angular image may be synthesised by generating p and q fields as in §5.3 and application of (6-10).

An angular image and its reconstruction according to this method are shown in Figure 6.3. The synthetic angular field (d) certainly appears more similar to the original than for the direct normal model (c) but is still not totally convincing. In particular, it appears more broken up than the original.

In order to model an angular field successfully, it appears that either a more complicated structure is required, or the auto-normal distribution must be discarded. Given the qualitative differences between model and data, this is inconvenient but perhaps not surprising. Below, we discuss a more viable alternative that is presently used.

6.3.2 Composite Feature Model

We shall relinquish the idea that features are independent (6-8) and attempt to model their joint distribution $P(\theta, c)$. The partial polar transform (6-10) may now be completed.

$$\begin{aligned}x &= c \sin \theta \\y &= c \cos \theta\end{aligned}\tag{6-12}$$

Although the convenience of being able to treat the Gabor features as independent, and hence of easily assessing their relative diagnostic effectiveness, has been lost, it is still possible to consider the random variables x and y independent. This assumption is likely to be a better approximation than for the orthogonal-normal components p and q because normalisation implied by (6-11) is avoided. We again use the estimation method of §6.2.1, finding \mathcal{G}_x and \mathcal{G}_y separately. Synthesis of new images proceeds as above, inversion of (6-12) is straightforward, and the structure of the Resultantsynthetic field is now much more similar to the actual data (Figure 6.3). This is also true even when the generating texture does not have a strong directional component. Similar trends are observed with Profile features (data not shown).

6.4 Implementation of the Classifier

Bayes classification (§3.4.1) is used to identify texture samples, with no reject option. Each class is assumed to be equally likely *a priori*, reducing the task of the classifier to selecting the class with the highest likelihood. In fact, the numerical value of the likelihood is usually extremely small, and it is more convenient to manipulate the log-likelihood. This has no effect on the result because log is a monotonically increasing function.

True-Likelihood Texture Classification

During classification, it is convenient to exploit the toroidal boundary approximation in a similar way to parameter estimation (§6.2.1):

$$\begin{aligned}2L_c(\mathbf{x}) &= \sum_{i \in \mathcal{C}} \log \lambda_i^{(c)} - MN \log(2\pi\sigma_c^2) - \sum_{i \in \mathcal{C}} \frac{|X_i|^2 \lambda_i^{(c)}}{\sigma_c^2} \\ \lambda_{m+nM}^{(c)} &= 1 - 2 \sum_{(k,l) \in \mathcal{N}^+} \beta_{k+lS}^{(c)} \cos \left[2\pi \left(\frac{mk}{M} + \frac{n\ell}{N} \right) \right]\end{aligned}\tag{6-13}$$

where $\lambda_i^{(c)}$ are spectral density coefficients computed from the parameter vector $\beta^{(c)}$, and $|X_i|^2$ are components of the test image power spectrum. The likelihood L_c is evaluated with parameters drawn from the MRF \mathcal{S}_c for each class in turn. As explained on page 149, the field mean is sometimes unreliable and is not always used during classification. When used, the MRF field mean may be subtracted from \mathbf{x} to centre the data; otherwise, the sample mean \bar{x} is used to remove the DC offset. True-likelihood classification is relatively fast under the toroidal boundary approximation because the image power spectrum $|X_i|^2$ and cosine terms used to form λ_i , may be pre-computed if sufficient memory is available — this is particularly valuable when matching one sample texture against many classes. Further advantage follows from the conjugate symmetry of power spectra, and the complexity of the true-likelihood classifier is no larger than for the FFT: $\mathcal{O}(MN \log M)$.

The form of (6-13) clearly implies that $\lambda_i > 0$. This breaks down if the homogeneity constraint is violated, which is often the case for parameter sets obtained by the pseudo-likelihood method (§5.2.3). In these circumstances, it is possible to drop the factor $|\mathbf{B}|$ from the likelihood because this appears in (6-13) as $\sum_i \log \lambda_i$, producing an approximate likelihood $L'_c(\mathbf{x})$ (Cohen & Cooper, 1987).

$$2L'_c(\mathbf{x}) \stackrel{?}{=} -MN \log(2\pi\sigma_c^2) - \sum_{i \in \mathcal{C}} \frac{|X_i|^2 \lambda_i^{(c)}}{\sigma_c^2} \quad (6-14)$$

Compare (6-13); L'_c also has the advantage of being faster to compute. However, this approximation is not taken further; we instead use the pseudo-likelihood classifier (6-15) which is unaffected by inhomogeneous parameter sets. Note that $P(\mathbf{x})$ is a probability density function because the model for \mathbf{x} is continuous. It is therefore quite possible to find $P(\mathbf{x}) > 1$ and hence $L(\mathbf{x}) > 0$.

Pseudo-Likelihood Texture Classification

There is no direct advantage of exploiting toroidal boundary conditions in the pseudo-likelihood function, which must be evaluated directly.

$$\tilde{L}_c(\mathbf{x}) = -\frac{MN}{2} \log(2\pi\sigma_c^2) - \frac{1}{2\sigma_c^2} \sum_{i \in \mathcal{C}} (x_i - \beta_c^T \boldsymbol{\eta}_i)^2 \quad (6-15)$$

This usually takes longer than for the true-likelihood classifier and has complexity $\mathcal{O}(MN\#\beta)$.

6.4.1 Sub-Optimal Classification Strategies

One may question whether the effort involved in evaluating the full Bayesian likelihood function for each possible texture class is worthwhile or whether a faster procedure would still give acceptable results. A number of approaches were discussed in §3.4: rather than comparing the test image with all class MRFs, it is possible to describe the image by its own MRF parameters and then to compare parameter sets. The simplest

way to compare sets of parameters is to treat them as vectors and compute the weighted Euclidean distance between them (§3.4.3; Manjunath & Chellappa, 1991).

$$\Delta_w(\mathcal{G}, \mathcal{H}) = \sum_j \frac{(\mathcal{G}_j - \mathcal{H}_j)^2}{\mathcal{G}_j^2 + \mathcal{H}_j^2} \quad (6-16)$$

We compared the accuracy of this method with true-likelihood Bayesian classification. With a ninth-order model, the weighted Euclidean classifier gave 44.7% error against 6.1% for Bayes. This increase is far too large to be tolerated, and the weighted Euclidean classifier was not used further.

More sophistication may be added to (6-16) by normalising the difference between each pair of parameters with a parameter variance (§3.4.3). This expression approximates the Bhattacharyya and Mahalanobis distance measures (3-8–3-10), equality holding if the parameters are normal and independent. It reportedly gives good results with Brodatz textures (Chellappa & Chatterjee, 1985; Kashyap *et al.*, 1982), but is inappropriate in the present circumstances: reliable estimates of the mean and covariances of the parameters must be available, but we have only a few training textures in each class, and so the required variances are not obtainable. Experience shows that the parameters covary strongly, which perhaps explains the failure of the weighted Euclidean classifier. Further, obtaining a true-likelihood parameter estimate for the test image is a relatively slow process, and could take as long as computing all the Bayesian likelihoods. This means that this method is potentially advantageous only when using pseudo-likelihood parameter estimates, which themselves introduce another source of error.

6.4.2 Classification using Gabor–Markov Models

When a single MRF is used to represent the feature array, as in the “direct normal model” (page 143), no special treatment is required. When two MRFs are needed, the overall log-likelihood is the sum of the parts:

$$L(x, y) = L(x) + L(y). \quad (6-17)$$

This simple form arises because the constituent fields and the corresponding sample data are assumed to be independent.

Feature images were classified using the orientation field alone, under the orthogonal-normal model, or using both orientation and contrast features with the composite feature model. Different results may be obtained with the Profile or Resultant feature-extraction methods, and some sets of results have been duplicated using both types.

Extraction of Gabor Features

Arrays of Gabor features were estimated from each image using either our Profile or Resultant extraction methods (§4.6), as indicated by the context. Only the LINE primitive was employed to extract Profile features. Both methods estimate orientation, contrast, and error features, but the latter were not used in this study (§5.3). Our experience guided the selection of a fixed set of Gabor filter parameters which was employed throughout. Following the notation of Chapter 4 (page 69), these parameters

were: ($T = 4$ pixels, $B = 1$, $\Delta\theta_{1/2} = \pi/6$, $\Theta = \pi/6$; $\sigma = 2.25$ pixels, $\lambda = \sqrt{3}$); hence each image was filtered at six orientations, two phases, and a single spatial frequency. Possible extensions to allow adaptive selection of filter parameters are discussed in §8.2.

6.4.3 Classification Procedure

The raw image material was 33 Brodatz images (Figure 1.6 on page 11), which were divided into quarters to give 132 independent gray-scale images of size 128×128 pixels. Not all the textures appeared very distinct to a human observer (e.g. #4 \leftrightarrow #9, #5 \leftrightarrow #28), and some of the photography could have been improved upon. No images were discarded, however, since these factors cannot always be controlled in practice, and the variation was taken to be representative. No pre-processing of the type described in §3.1.3 was performed.

Two variants on the “leave-one-out” procedure were used to generate texture classes. Under the *All-Quarters* method, a separate MRF was estimated for each of the 132 images. During classification, each image was then matched against the MRFs of the remaining 131 images. The result was a “success” if the match was with another quarter of the same parent image. This method is potentially susceptible to inhomogeneities in the image, since one errant quarter may indeed be more similar to part of another parent texture! The more conventional *Group* method was to form 33 separate MRFs, taking all quarters of one parent together. During classification, each sample was matched against the 32 classes to which it did not belong, and against a new MRF estimated from the remaining three quarters of the same parent. Consequently, test and training image quarters were always disjoint.

Both our methods derive a texture class from a single parent image. The number of classes is larger than that normally considered (e.g. Chellappa & Chatterjee, 1985; Cohen *et al.*, 1991; Vickers & Modestino, 1982), making correct classification more difficult.

Treatment of Field Mean

Classification was performed both with and without the MRF mean. In the latter case, the test image was simply adjusted to have zero mean. Making use of the mean for a pixel image may be thought of as “cheating”, since it exploits atypical consistency in our imaging environment. Lighting and camera parameters may normally be expected to vary for different exposures, making this information unreliable. In any case, these parameters had been adjusted to make good use of the dynamic range offered by the camera, so the pixel means were very similar. The first step in extracting Gabor features from an image is to eliminate the DC component (§A.2.4), and so the *feature* mean is not covered by the above comments: it is a valid texture property. Both sets of results are given below (with and without mean), but a fair comparison is between the pixel classifier without mean and the feature classifier with mean.

Pixel variance is another first-order statistic, and is similarly unreliable (§3.1). Changes of illumination affect the absolute variability of the texture, which is reflected in the MRF variance σ^2 , but the *relative* variability is unchanged. Global changes in illumination could have been compensated for by scaling the MRF variance (Kashyap

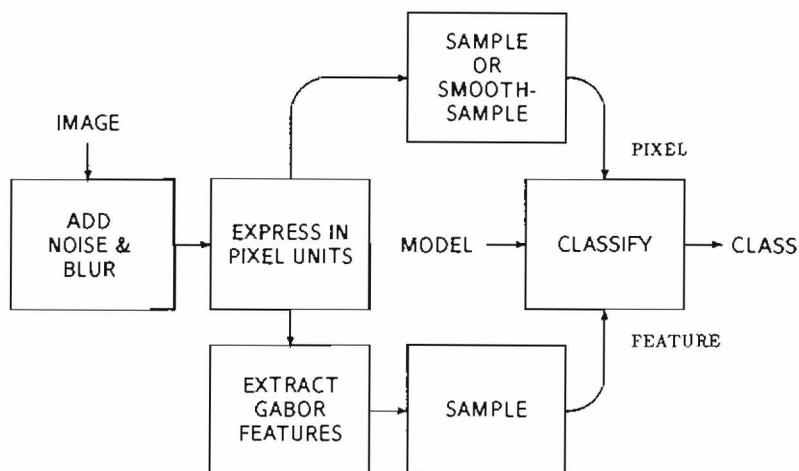


FIGURE 6.4: **Classification procedure.** Images are classified according to the procedure illustrated above. When required, image corruption occurs prior to sampling and feature extraction. No sampling is performed by the conventional pixel classifier.

et al., 1982), or by a modified classifier (§5.4); local changes by logarithmic filtering (Voorhees & Poggio, 1987). Neither procedure was adopted in our experiments.

Evaluation of Classifier Robustness

When assessing the tolerance of the classifier to image corruption, only the test images were altered: the same training data were used. Robustness is important in a practical environment since imaging parameters cannot always be fully controlled. The procedure for pixel and feature classifiers is illustrated in Figure 6.4. Zero-mean white random noise was added to the image before processing. Pixel values were quantised and clipped to the permitted range. This implies that noise and image data were not fully independent; for example, noise cannot decrease the intensity of a fully-black pixel. The noise-generator was non-deterministic and hence classification accuracy may differ slightly for different runs.⁴ The amount of noise is characterised by its variance ρ^2 , measured in gray-level units squared.

Blur was modelled rather ideally by convolution of the image with a circularly-symmetric normalised Gaussian kernel. No anti-aliasing was done, causing the filter to have virtually no effect when the intended amount of blur was very small. Clipping could not occur, but the process was irreversible because the blurred image was quantised to pixel units. The amount of blur was characterised by the space-constant

⁴This was a minor effect, however, and did not disguise the trends noted below.

of the Gaussian, b , measured in pixel units. Additive noise and blur were not applied simultaneously.⁵

6.5 Performance of Texture Classifiers

Several aspects of classifier performance are of interest. The chief comparison is between conventional pixel models and our new Gabor–Markov feature models: we wish to compare trade-offs between accuracy, speed, and robustness; and to see whether either type of model is better suited to a particular class of images. In order to make a fair comparison, we investigated the influence of model order, and the use of true or pseudo likelihoods. For the feature models, there is the further question of what level of sampling may be tolerated and which feature-extraction and modelling procedures should be used. Texture classification gives a simple quantitative performance measure for each combination.

In order to assist comparison between different sets of results, graphs of classifier accuracy are collected together at the end of this Chapter (pages 165–173).

6.5.1 Basic Results

Accuracy of Conventional Pixel Classifier

True-likelihood classification was performed using the All-Quarters method (§6.4.3) with MRF parameter sets of order 1–9. The first-order model has a tiny neighbourhood set containing only four pixels (Figure 5.1 on page 107), and this is clearly insufficient to capture the structure of the different textures (Figure 6.10 on page 165). A third-order model has only seven parameters, yet the mis-classification rate is only 7%. As neighbourhood size is increased further, there is only a marginal further improvement in accuracy: a ninth-order model with 25 parameters achieves a 6% error rate. This difference is hardly significant and the additional computational effort does not seem worthwhile. Including the MRF mean has a marginal beneficial influence on classification accuracy but, as discussed above (page 149), this parameter is unreliable in general.

Using the Group method gives a very similar pattern of results. Accuracy is slightly impaired: the error rate rises from 7% (All-Quarters) to 8% (Group) for a fifth-order model ignoring field mean. The distinction between different classification methods seen in Figure 6.10 is repeated for most other image conditions. In order to enhance clarity, only the All-Quarters results, without field mean, are shown graphically in most cases.

Pseudo-likelihood classification is less accurate (Figure 6.11), achieving a minimum error rate of 17.4% without field mean — this rate was reached with a third-order model. Increasing the model complexity further tends to decrease accuracy. There is a sharp jump in error rate for a second-order model, for which we do not have a convincing explanation. The effects of field mean and classifier design are similar to those for the true-likelihood classifier. Generally, the pseudo-likelihood classifier is inferior: there is

⁵Some quantisation noise arises when the image is blurred, however.

no speed advantage when the toroidal boundary approximation is fully exploited and when parameter estimation is excluded, and accuracy is reduced. Accordingly, we have concentrated our resources on true-likelihood classification, noting comparisons with pseudo-likelihood where appropriate.

Accuracy of Gabor-Markov Classifiers

A putative advantage of using features to represent the image data is that they can be sampled more sparsely than the pixel array, thereby decreasing the computational requirement (§5.7). In a complete Gabor representation (§4.1.4; Porat & Zeevi, 1988), the sampling period T_s is the same as the period of the trigonometric envelope of the filter kernel (4.16 on page 62). For our filter parameters (§6.4.2), this gives $T_s = 4$ pixels, allowing the quantity of data to be reduced by 93% from 128×128 to 32×32 pixels. It is likely that the image will have to be sampled more densely than this, however, because we took only one layer of the Gabor pyramid and hence cannot achieve a loss-less representation.

This comparison is not entirely fair because the present implementation represents Gabor features using floating-point (4 bytes) compared to 1 byte for pixel data. This extra precision is not necessary, however, and a scaled single byte per feature would suffice. An unsampled two-feature representation would then require twice as much storage as the original image, dropping to 12.5% after sampling.

Our Brodatz image set was classified using both orthogonal-normal and composite-feature models, for Profile and Resultant features. Accuracy using the orientation field alone, with the orthogonal-normal model, is shown in Figure 6.12 as a function of sampling period s for a fifth-order model. Using the field mean improves accuracy considerably, as would be expected, but there is surprisingly little effect of the degree of sub-sampling. Error rates for this method are higher than for the pixel classifier.

The sampling period again has little effect on accuracy when using the composite-feature model and Resultant extraction method (Figure 6.13). When the field mean is used, accuracy rises as the sampling period is increased. Model order has only a small impact (Figure 6.14); highest accuracy is obtained for a model of intermediate complexity. For the most favourable choice of parameters, accuracy is virtually the same as for the pixel classifier (6.0% against 5.3% error) but this is achieved with a field reduced to *one-sixteenth* of its original area. When the field mean is not used, error rates are considerably higher, exceeding those obtained by the pixel classifier.

A similar pattern is observed for the Profile extraction method (Figure 6.15). Maximum accuracy is achieved for a sampling period of three pixels with the AllQuarters method, and for a greater period with the Group classifier. There is virtually no variation of accuracy with model order between second and ninth order (data not shown).

The composite feature model uses both orientation and contrast features, and is consistently more accurate than the orthogonal-normal model derived from orientation alone. Since two random fields are used to represent both models, there is no significant computational penalty for using the composite-feature model. We shall be chiefly concerned with this model below, commenting on the orthogonal-normal model only in passing.

There is no clear choice between the Profile and Resultant feature-extraction methods: they require similar amounts of computation and offer similar performance. We shall concentrate on the Resultant method because it appears to have slightly higher accuracy for unsampled and heavily-sampled fields, and is marginally faster to calculate. Our discussions below of the “feature classifier” will refer to this model, using true likelihood. Preliminary investigations of the use of pseudo likelihood with feature data suggested a similar pattern to that achieved for pixel data, classification accuracy being far less accurate than with true-likelihood parameter sets.

6.5.2 Classification of Noisy Images

Noise is unavoidable in real images. Our noise model (page 150) is idealised but is valid for comparative purposes, giving a measure of the relative robustness of classifier types. The classification task ensures that models which are either too specific or too imprecise are penalised.

Accuracy of Conventional Pixel Classifier

Additive random noise was applied to the image set prior to classification, as described in §6.4.3. The variance of the noise is measured in gray-level units squared. As may be seen immediately from Figure 6.16, the pixel classifier is extremely sensitive to noise, particularly for models with many parameters. Even noise of variance $\rho^2 = 3$ causes the error rate to triple for a fifth-order model; for noise of variance 20 or greater, more than half the corrupted textures are incorrectly classified, compared to an original error rate of 6%. This surprisingly sharp drop in performance undermines the utility of this classifier. As can be seen from the sequence in Figure 6.5, the texture is still visually easily recognisable after the addition of much larger amounts of noise than this, suggesting that the classifier is making poor use of the available information.

Another trend present in the results is for a classifier of larger order to be *more* seriously affected by noise. This reverses the ranking for uncorrupted images. It might have been supposed that the larger neighbourhood size of the model of higher order would be more effective in suppressing the noise, but this is clearly not the case. A third-order model gives highest accuracy with most noise levels. Possible causes for this effect are discussed in §7.3. As before, Group classification shows a similar trend but accuracy is marginally reduced (data not shown); using the field mean gives a slight improvement. Noise is even more destructive for the pseudo-likelihood classifier. A fifth-order model achieves 17.4% error for no noise, rising sharply to 78.0% for $\rho^2 = 3$ (data not shown). Such high error rates mean this classifier is virtually unusable for noisy images.

There is no direct precursor to these results in the literature. When noise is considered, it is in conjunction with regions of constant luminance rather than a random texture (Geman & Geman, 1984; Hansen & Elliott, 1982; Lakshmanan & Derin, 1989). This is a qualitatively different problem to that faced here — classification is not really an issue with these primitive textures, since the best available estimate is computed directly from the sample mean.

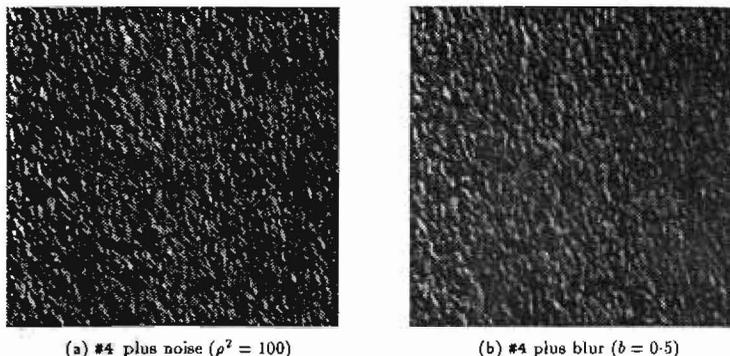


FIGURE 6.5: Effect of noise and blur on Brodatz image. The pressed cork image (#4; Figure 1.6 on page 11), 256×256 pixels, was corrupted by the methods used to test classifier robustness (page 150). (a) addition of white noise, variance $\rho^2 = 100$ gray-level units squared. (b) blur with parameter $b = 0.5$ pixel units. In both cases, the original texture is still sufficiently discernible to identify it visually. This suggests that high error rates from the pixel classifier are caused by a poor use of the available information.

Accuracy of Gabor–Markov Classifiers

Given that Gabor filters have a region of support extending over surrounding pixels, we would expect features to be less affected by noise than the pixel values themselves. Consequently, the performance of the feature classifier should improve relative to that of the pixel classifier as the level of noise is increased. This supposition is strongly supported by our results (Figure 6.17): the All-Quarters fifth-order model with a sampling period of 4 pixels achieves an error rate of 7.5% for no noise, increasing to 8.3% for $\rho^2 = 20$. This compares favourably to error rates of 6.8% and 52.4% for the conventional fifth-order pixel classifier. A roughly linear increase in error with noise variance is seen in both cases but the slope is lower by a factor 45 for the composite-feature classifier. The accuracy of the feature classifier declines only gradually with higher noise levels, and also receives the computational benefits of sub-sampling. As may be seen in Figure 6.17, sampling more heavily decreases both the slope and intercept, as well as reducing the computational load.

When the feature mean is ignored, the error rates are a little higher but the trend is the same (data not shown). Unlike the pixel MRF, the order of the feature model has only a weak influence on accuracy between third and ninth order. In both cases, the additional complexity of implementing a full ninth-order model is not justified: fifth order is a better choice. Error rates with the orthogonal-normal classifier are consistently higher than for composite features (data not shown).

6.5.3 Classification of Blurred Images

Camera blur and pixel leakage is another pervasive potential source of image corruption. We model this by Gaussian blur with parameter b , measured in pixel units (page 150). Only very small values of b were taken, and hence the actual effect on the image is somewhat complicated by aliasing. Pixel and feature classifiers receive the same treated images, however, and so their comparative performance is still valid.

The pixel classifier was hardly affected by a small amount of blur ($b = 0.25$ pixel), possibly because of aliasing, but was virtually destroyed by a larger amount ($b = 0.5$), the error rate exceeding 80% (Figure 6.18). There is little effect of model order — this is surprising because a larger neighbourhood may have been expected to respond on the basis of remaining medium-scale structure. The pixel-based classifier is almost unusable with blurred images, because its error rate is only a little better than chance (97.7%).

Blur is less detrimental to the feature classifier (Figure 6.19): the fifth-order composite-feature method gives an error rate of 54% for $b = 0.5$ against 84% with the pixel classifier. Both the baseline error and rate of increase are lower for increased sampling period. It is not meaningful to quantify this difference in terms of a change of slope because too few points are available. There is a mild effect of model order (data not shown) favouring more complex models; a fifth-order model is a good compromise. Errors are higher if the field mean is discarded or if the Group classifier is used (data not shown).

In view of its inferior noise tolerance, a surprising result is that the orthogonal-normal orientation classifier is less sensitive to blur than the composite-feature method. Both sets of data are shown in Figure 6.19. With a sampling period of four pixels, the error rate for the fifth-order orientation classifier after blur with parameter 0.5 was only 17.4%. It appears that the orientation feature is virtually unaffected by this amount of image blur. The orientation field extracted from the pressed cork image (#4; Figure 1.6 on page 11) does indeed appear to be slowly-varying (Figure 6.3). Unfortunately, the orientation classifier is more sensitive to noise and has a higher baseline error, and so the composite feature classifier is still the best choice overall.

6.5.4 Comments and Conclusions

It is clear from the above results that many of the anticipated benefits of a switch from pixel- to feature-based classification do occur in practice:

- the feature array may be sampled to $1/16$ of its original area, bringing considerable computational benefits, without loss of accuracy;
- the composite-feature model is up to 45 times more noise-tolerant than the pixel classifier;
- blur tolerance is also increased, particularly with the orthogonal fields model.

There is no subset of our Brodatz images for which either method is particularly well-suited: errors are similarly distributed. This is illustrated by the confusion matrices

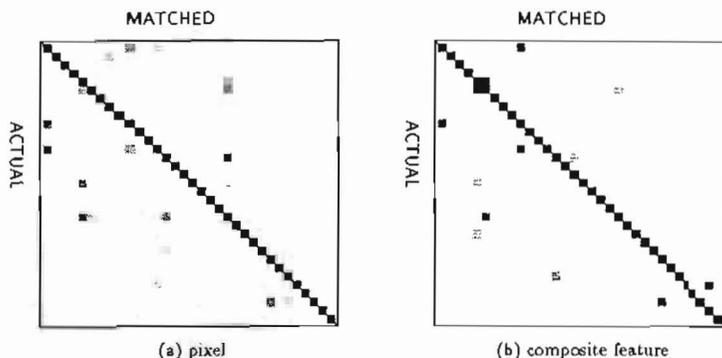


FIGURE 6.6: Scatter matrices for pixel and composite-feature classifiers. Our Brodatz image set was classified using true-likelihood parameter sets by the All-Quarters method. Matrix order follows the numerical labelling of the images (Figure 1.6). Correctly classified images lie along the leading diagonal. Intensity is proportional to frequency. (a) pixel classifier for models of order 2–5, 7 and 9. (b) Resultant composite-feature classifier for models of order 2, 5 and 9. Almost all errors occur with the same images for both classifier types.

shown in Figure 6.6. Misclassification errors, shown by off-diagonal entries, usually occur on the same row for both pixel and feature classifiers. Equal accuracy is obtained by the two methods for uncorrupted images. The greater robustness of the feature method is important as it may not always be possible in practice to obtain images of high quality.

With both classifier types, it is clear that a ninth-order model does not offer sufficiently improved performance relative to the more economical fifth-order model to justify its use, and in many cases the ninth-order classifier was *less* accurate. A second-order model is too small to distinguish textures reliably, and the ideal choice lies between third and fifth order. A sampling period of four pixels is most advantageous with the composite-feature method, tending to reduce both computational load and error rate. Unless blur is a particularly serious hazard, the composite-feature model, derived from both contrast and orientation features, is superior to the use of orientation alone.

Our conclusions are based mainly on observations of accuracy using the All-Quarters method. Very similar trends are observed with more conventional Group classification, however, confirming that our results are genuine. Initial results with the pseudo-likelihood classifier suggest that it is even more sensitive to noise than true likelihood. When used with the pseudo-likelihood function, the sampled feature classifier again achieves comparable accuracy to the pixel classifier with uncorrupted images (data not shown). In these circumstances, greater immunity to image corruption may be more valuable than with true-likelihood classification.

With the current serial implementation, there is a significant computational cost involved in representing the image by Gabor features. Mainly simple local computation is required, however, and a suitable parallel machine or special-purpose architecture could

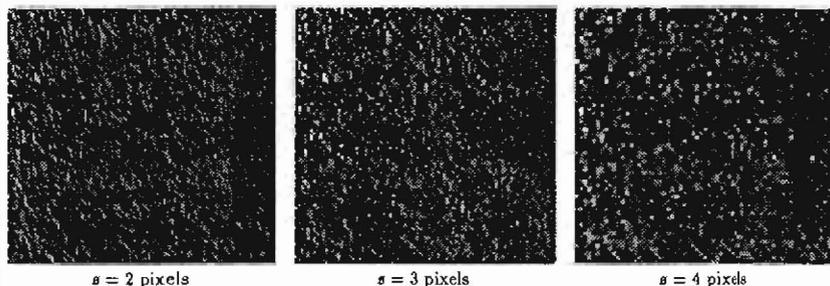


FIGURE 6.7: Sub-sampled Brodatz image. Sub-sampled versions of the pressed cork image (#4; Figure 1.6 on page 11) formed by taking every second, third and fourth pixel, respectively. Pixel size has been adjusted so that each printed image occupies the same area. When used with the pixel classifier, a sampling period of two or three pixels leads to higher accuracy than with unsampled images.

easily be constructed to perform this task (§4.1). Using this method, the processing time of the feature extraction stage could be made small compared to classification or parameter estimation.

6.6 Performance of Sampled-Markov Classifiers

The twin advantages of using Gabor features are robustness and sparse sampling. From the experiments of the previous section, it is not clear whether these are conferred by the nature of the Gabor features themselves, or simply because of an increased area of support for each feature point. Sub-sampling has a strong influence on computational requirements, achieving a reduction of an order of magnitude, and is therefore of immediate practical interest. In order to assess the influence of increased region of support and of sub-sampling on the feature classifier, we investigated the effect of building them into a modified pixel-based classifier.

6.6.1 Classification using Sub-Sampled Markov Models

Sampling with period s pixels was achieved by retaining only the first pixel from each $s \times s$ block in the image — this immediately reduced the volume of data by a factor s^{-2} . Missing parts of boxes were ignored. Only integer s was used, but sampling with a suitable weighting to avoid aliasing could easily overcome this restriction (§8.3). No attempt was made to control aliasing errors. This crude sampling method mirrors that used to sample the output of the feature detector. Since the structure of the sampled images differed from their unsampled counterparts, fresh sets of MRF parameters were obtained. The visual effect of sampling is illustrated by Figure 6.7: there is surprising resilience for a sampling period of two pixels but image quality declines significantly thereafter.

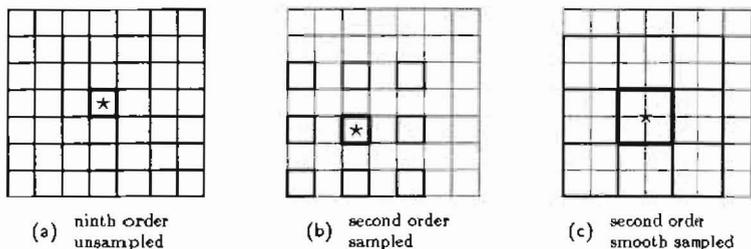


FIGURE 6.8: Comparison of original and sampled Markov neighbourhoods. Illustration of MRF neighbourhood structures, centre pixel marked. (a) unsampled ninth-order (Figure 5.1 on page 107); (b) sampled second-order, period 2 pixels; (c) smooth-sampled second-order, period 2 pixels. Both sampled neighbourhoods project into a smaller image “footprint” than the original ninth-order model.

Classification of Uncorrupted Images

The effect of sub-sampling is shown in Figure 6.20. Remarkably, discarding most of the data by sub-sampling *reduces* the error rate — from around 6% for the dense image to 0.7% when $s = 2$! Accuracy is raised for models of all order when the sampling period is two or three pixels but declines for heavier sampling. A period of two pixels gives minimum error. Given the perceived decrease in image quality produced by sampling, this improvement in accuracy is startling. It seems highly counter-intuitive that discarding nearly 90% of the data should *improve* classifier performance!

There are several possible explanations for this effect:

1. smaller image size leads to a less-complicated likelihood function, allowing gradient ascent to find a better maximum;
2. when projected back into the dense image, the span of the MRF neighbourhood is increased, allowing the model to reflect larger-scale correlation;
3. the sampled image achieves a closer fit to the auto-normal model, thereby increasing efficiency.

Failure of the parameter estimation algorithm appears not to be the cause, ruling out our first suggestion. This conclusion is supported by the appearance of similar trends for both true and pseudo-likelihood classifiers, even though the latter is unaffected by local likelihood maxima. The second possible cause is due to increase in the image area covered by the neighbourhood set of the sub-sampled model when this is projected back onto the original. Although only some of the values within this neighbourhood are present, it could be suggested that this is more than offset by the availability of longer-range information. This hypothesis is not confirmed by the data. The projected neighbourhood of the sub-sampled second-order model ($s = 2$ pixels) is contained within that of the full ninth-order model (Figure 6.8), yet it still achieves higher accuracy. The third suggestion is more intriguing; we return to this below (§7.1.3).

Just as the coding estimate is inefficient because it uses only a small proportion of the available data, the sub-sampled classifier appears to be very wasteful (Besag, 1974; Cross & Jain, 1983). Another similarity with codes is that a shift in the arbitrary sampling origin yields a different set of data. We have not sought to compare or combine these separate estimates (preferring instead to use the smooth-sampling method, §6.6.2). Whereas a code retains the whole image but uses only part of the conditional probability structure, sub-sampling discards most of the data entirely.

As has previously been the trend, classification accuracy was marginally improved by using the pixel mean, and reduced a little by the Group method (data not shown), but similar patterns of results were obtained. Increasing the order of the MRF model beyond fifth-order did not convey any real benefit to classification performance. We shall present the results for corrupted images for a fifth-order model. This level of complexity achieves a reasonable compromise between speed and accuracy, and the results follow a typical pattern.

Classification of Noisy Images

Sub-sampling cannot counteract noise directly because there is no pooling of data from neighbouring pixels. One would predict the sub-sampled classifier to be no more noise-tolerant than the original, and possibly less so, because a reduction in effective image size diminishes scope for cancelling out random fluctuations.

It is immediately clear from the results for a fifth-order MRF (Figure 6.21) that this is not the case: the sub-sampled classifier is massively *more* tolerant to noise than the original version (shown dotted). Whereas the error rate for the full pixel classifier exceeds 50% for noise of variance $\rho^2 > 20$, the sub-sampled classifier with sampling period $s = 3$ pixels has an error rate of only 5% for noise of variance $\rho^2 = 20$ rising to 11% for $\rho^2 = 50$. Increasing the degree of sampling decreases the noise-error slope further.

These results are highly counter-intuitive. A fifth-order MRF was discussed above as a good compromise between speed and efficiency, but the same pattern is observed for models of different order (data not shown). It is easy to construct quite different textures that would be difficult to distinguish on this basis but our image set was chosen without any covert selection criteria and presumably does not contain any of these pathological cases.

The familiar advantage of using the pixel mean and disadvantage of using the Group method were again observed (data not shown). They did not appear to interact greatly with the variance of the noise. Unlike the full classifier (Figure 6.16), accuracy did not appear to decline for models of higher order for the moderate noise levels examined, reaching a plateau between third and fifth order.

Classification of Blurred Images

Even a small degree of blur was sufficient to destroy the performance of the unsampled pixel classifier. Although blur does still have a serious adverse impact on the sub-sampled classifier, this is less severe than before (Figure 6.22). The information

conveyed by the pixel mean is very helpful for large amounts of blur ($b = 0.5$ pixels) but even without this, an error rate of 26% is achieved with a sampling period of 3 pixels compared to 84% previously. We observed a small increase in accuracy for higher order; this trend could also be seen for the Group classifier (data not shown).

It is clear from Figure 6.22 that the rate of error increase with blur is consistently reduced for heavier sampling. This interaction is similar to that observed for additive noise. Very heavy sampling leads to an increase in baseline error (no noise or blur) suggesting that the best classification method should be chosen with regard to the anticipated level of image contamination.

Summary of Results using Sub-Sampled Markov Models

There is a strong computational incentive for reducing the amount of data used to represent each image but one would expect this to be balanced by a performance penalty. Remarkably, however, sub-sampling *increases* classification accuracy: a sampling period of two or three pixels gives fewer errors than the dense classifier with uncorrupted images. Noise and blur tolerance were increased, errors rising more slowly the larger the sampling period. Minimum error for uncorrupted images was seen with a sampling period of two pixels, so a trade-off does operate between baseline accuracy against robustness and computational efficiency. Unsampling fields achieve a very poor balance, offering low baseline accuracy, minimum robustness and minimum efficiency. The best compromise is a fifth-order model with a sampling period of two or three pixels.

6.6.2 Classification using Smooth-Sampled Markov Models

The large performance increases seen in the previous section are even more surprising when one considers the drastic aliasing effects of the primitive sub-sampling method. Although still far from optimal, an improved technique is to average the pixels in each box (Meer *et al.*, 1987); no weighting was used, and the boxes did not overlap. We characterise this smooth-sampling by the factor by which the side of the image is reduced; hence a sampling frequency $f = 0.5$ reduces the area by 75%. The resulting image is expressed in gray-level units, rounding any fraction, and any remaining fractions of a box are discarded. Only integer values of f^{-1} were used. In the discussion, we shall take “s” to imply sub-sampling and “f” smooth-sampling. Full anti-aliasing, which allows the original texture to be reconstructed from the sampled data, is discussed in §8.3.

All the data is used in this type of sampling, suggesting that efficiency should be higher than sub-sampling alone. Shifting the sampling origin still changes the sampled image, but aligning the sampling box with the image borders is the most natural choice.

Classification of Uncorrupted Images

Smooth sampling reduces the error rate for uncorrupted images below that of the sub-sampled classifier (Figure 6.23). At $f = 0.5$, a fifth-order classifier achieves an extremely low misclassification rate of 0.7%, falling further to 0% if the pixel mean is used. This compares with 1.5% (0.7%) for the corresponding sub-sampled classifier ($\varepsilon = 2$ pixels),

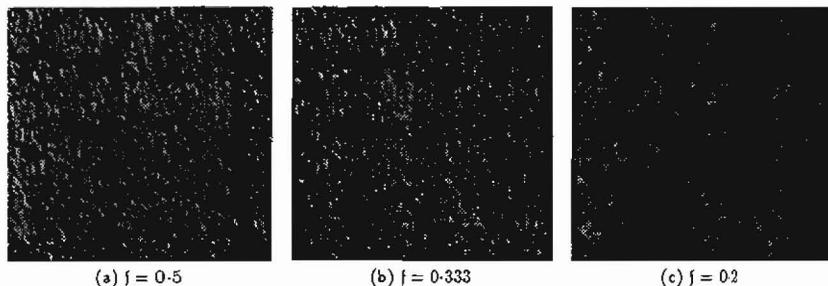


FIGURE 6.9: **Smooth-sampled Brodatz image.** Smooth-sampled versions of the pressed cork image (#4; Figure 1.6 on page 11) formed by averaging over a box of size: (a) 2×2 pixels, (b) 3×3 pixels, (c) 5×5 pixels; and sampling with frequency f ($1/\text{pixel units}$). Pixel size has been adjusted so that each printed image occupies the same area. When used with the pixel classifier, each of these sampling periods gives higher accuracy than with the unsampled image set.

and 6.8% (6.8%) by the original method. Smooth sampling is consistently more accurate than sub-sampling when the same volume of data is retained. A high degree of sampling is possible — the classifier for sampling frequency $f = 0.2$ achieves an error rate of 4.5% (3.7%) with a set of data reduced to 4% of its original size. Minimum error is achieved with a sampling period of two pixels (only integer values were tested). When the sampling period was increased to five pixels ($f = 0.2$), accuracy is comparable to that for the dense image.

A graph of accuracy against MRF order is again L-shaped, reaching a plateau between third and fifth order (Figure 6.23). The single error at $f = 0.5$ at fifth or seventh order occurs when a quarter of #2 (fieldstone) is classified as #73 (soap bubbles). For the same parameters, the Group classifier makes five errors. Given the fine variation present in the original textures, it seems remarkable that good accuracy can be achieved after spatial averaging. Smoothly-sampled versions of #4 (pressed cork) are shown in Figure 6.9: averaging has reduced the dynamic range noticeably (compare also Figure 6.7 on page 157). Both sets could be made more faithful to the original by better anti-aliasing but this might jeopardise accuracy improvements.

Sampling also improves the accuracy of the pseudo-likelihood classifier although it is still less reliable than true likelihood. A fifth-order pseudo-likelihood model sampled every two pixels has an error rate of 5.3% after sub-sampling, and 6.8% after smooth-sampling, compared to 17.4% for the dense image. Unlike the true-likelihood classifier, averaging appears to decrease accuracy.

Classification of Noisy Images

We would expect the smooth-sampled classifier to have good noise-tolerance because it pools data from a region, thereby reducing the effective noise variance by a factor f^2 . For very small f , this effect fails because the remaining data will be insufficient to

characterise the texture. Because the sampled image is not re-normalised to use the full pixel range, the amount of information diminishes faster than the size of the processed image.

Classification accuracy for various levels of noise variance ρ^2 and degrees of sampling are shown in Figure 6.24; $f = 1$ (dotted) corresponds to the un-processed image. There is a dramatic difference in noise tolerance: the rate of increase of error with noise variance is approximately 200 times smaller with the smooth-sampled classifier compared to the original unsampled method. An error of only 8% is achieved for noise of variance $\rho^2 = 200$ and smooth-sampling frequency $f = 0.2$. Model order has a very weak influence on classifier accuracy between third and ninth order, but this was difficult to judge accurately because error rates were so low.

This pattern of results is very similar to the sub-sampled classifier: a substantial improvement in noise-tolerance compared to the original method, with a reduced slope but higher intercept for larger sampling period. The smooth-sampled classifier is more accurate: for noise $\rho^2 = 50$ and sampling period 2 pixels, it achieves an error of 3% compared with 20.4% for a corresponding degree of sub-sampling. Similar trends are again observed for the Group classifier, and when the field mean is used (data not shown).

Classification of Blurred Images

As predicted from the large area of support for the smoothly-sampled image, the effect of blur is reduced as the sampling period is increased. In all cases shown, the new classifier out-performs the conventional unsampled classifier (Figure 6.25): for moderate blur, $b = 0.5$ pixels, the old method achieved an error of 84.0% (83.3% with mean) compared to 5.3% (4.5%) for smooth-sampling at $f = 0.2$.

Contrary to the trend for additive noise, models of larger order are more accurate for unsampled blurred images (§6.5.3), and this is also true after sampling. Accuracy is improved when the field mean is used, but is lower for the Group classifier although similar trends were observed (data not shown).

Summary of Results using Smooth-Sampled Markov Models

Smooth-sampling with a flat window function provides a degree of anti-aliasing but is far from ideal in this respect. Nonetheless, performance advantages of the primitive sub-sampled classifier (§6.6.1) were extended further. Using the window function and a sampling period of two pixels, extremely high accuracy is achieved: zero error when the field mean is used. This compares with 6% for the dense classifier. Tolerance to noise and blur were both extended, surpassing the levels achieved by Gabor-Markov classifiers. A trade-off was noted: a sampling period of two pixels gives highest baseline accuracy but heavier sampling offers greater robustness and computational efficiency. The dense classifier scores poorly with this price-performance equation, giving minimum reward for maximum cost. A fifth-order model with a sampling period of two or three pixels is a good general choice, but a larger sampling period should be considered if processing time is critical and an adequate volume of data is available.

6.7 Comparison of Texture Models

Building on Chapters 4 and 5, we proposed a novel method of texture analysis, modelling the spatial distribution of Gabor features with Markov random fields. Several members of the new class of Gabor–Markov models were proposed. We also proposed a Sampled–Markov framework, which employs representations derived from sampled images. For reasons of computational tractability, we adopted the auto-normal MRF for conventional, Sampled–Markov and Gabor–Markov models. Using Bayesian texture classification as a benchmark, we compared the performance of our new Gabor–Markov method with that of the conventional MRF treatment. Our motivation in proposing the Gabor–Markov model was to seek enhanced robustness through greater abstraction, and higher computational efficiency. Both objectives were achieved: similar accuracy was attained by Gabor–Markov and conventional classifiers for uncorrupted images, with a minimum error rate of 6%, but our new model derives efficiency benefits from a reduction of effective image area of up to 94% (Figure 6.26). Tolerance to noise was increased by a factor of up to 45, and the orientation classifier was virtually unaffected by blur. Since blur and noise are unavoidable properties of real images, these robustness improvements are of practical importance. There was little distinction in accuracy between the Profile and Resultant feature-extraction methods (§4.6), and similar trends were seen for All-Quarters and Group classifiers, and for true and pseudo likelihoods.

Our experiments with the conventional MRF algorithm led to proposals for Sampled–Markov models. Pixel selection occurs immediately for the Sub-Sampled classifier; after first smoothing with a simple window function for the Smooth-Sampled version. Primitive sub-sampling was not expected to be a good way of reducing image area: it discards important high spatial-frequencies and is prone to aliasing error. In the present context, these objections could be overcome by instead taking a smaller contiguous patch of the texture since the image is assumed to be homogeneous. We found, however, that sampling has a far more beneficial effect than we had initially supposed.

Image sampling improves classifier accuracy, reliability and efficiency. The greatest benefits are seen with the Smooth-Sampled classifier: this achieves the highest accuracy of any classifier for uncorrupted images, with a misclassification rate of less than 1%. Tolerance to noise is increased relative to the unsampled images by a factor of up to 200, and is also superior to the feature classifier (Figure 6.26); blur tolerance is considerably higher than for dense images, and again exceeds that of the feature classifier (Figure 6.27). These performance improvements are obtained in conjunction with the computational benefits of sampling, giving the smooth-sampled classifier optimal performance in all respects. Preferred parameters are: fifth-order, true likelihood, with a sampling period of between two and three pixels. A sampled classifier is inappropriate when the image area is already small because too few pixels would remain after sampling to characterise the texture reliably. The feature classifier is similarly affected, and the dense pixel classifier should be used in these circumstances.

These results are of immediate practical interest because they suggest that both Gabor–Markov and Sampled–Markov methods offer improved performance at lower computational cost than a conventional Markovian analysis. Our experiments were performed on natural textures drawn from the popular Brodatz album. Performance trends

are derived from over 110 000 classification results, and appear sufficiently clear-cut to suggest their wider validity. The success of Sampled-Markov classifiers is remarkable because they initially appear to make very inefficient use of the available information. In Chapter 7, we assess factors influencing classification accuracy, discuss possible causes for the observed trends, and propose modifications leading to further improvements in accuracy and robustness.

6.7.1 Summary of Achievements

In this Chapter, we have:

- proposed several members of a new class of hybrid Gabor–Markov models;
- used texture classification accuracy as a benchmark to demonstrate superior performance of our novel approach compared with conventional Markovian techniques, including — reduction of effective image area by a factor 16 without loss of accuracy, improvement of noise tolerance by up to 45 times, considerable improvement of blur tolerance;
- proposed a new family of Sampled-Markov models, and shown that they outperform conventional dense models by an even greater margin, including — optimal error rate of less than 1%, reduction of image area by a factor 25, noise tolerance increased by up to 200, vast increase in blur tolerance.

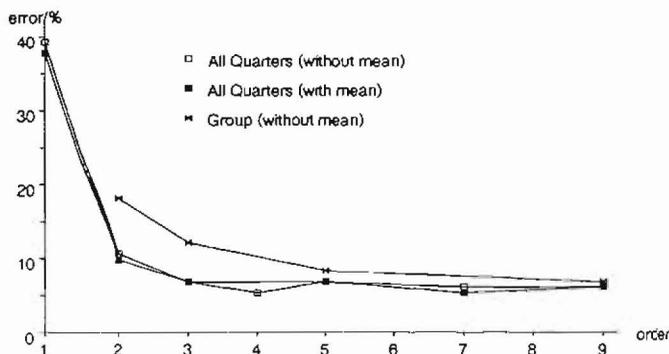


FIGURE 6.10: Accuracy of unsampled pixel classifier. Error rate against MRF order for the unsampled uncorrupted true-likelihood pixel classifier. 132 Brodatz image quarters were classified as described in §6.4.3, using All-Quarters and Group methods.

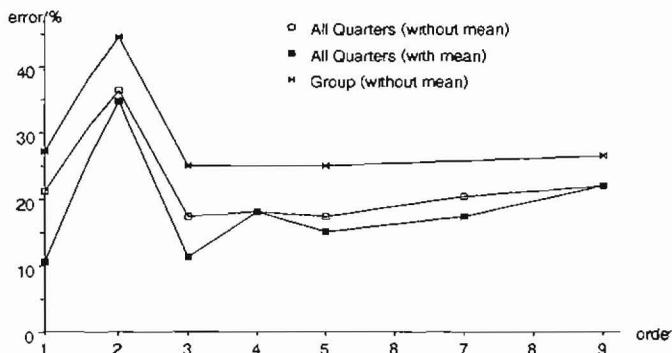


FIGURE 6.11: Accuracy of unsampled pseudo-likelihood pixel classifier. Error rate against MRF order for the unsampled uncorrupted pseudo-likelihood pixel classifier. 132 Brodatz image quarters were classified as described in §6.4.3, using All-Quarters and Group methods. True-likelihood classification is more accurate (Figure 6.10). The low error-rate for the first-order model is unexplained.

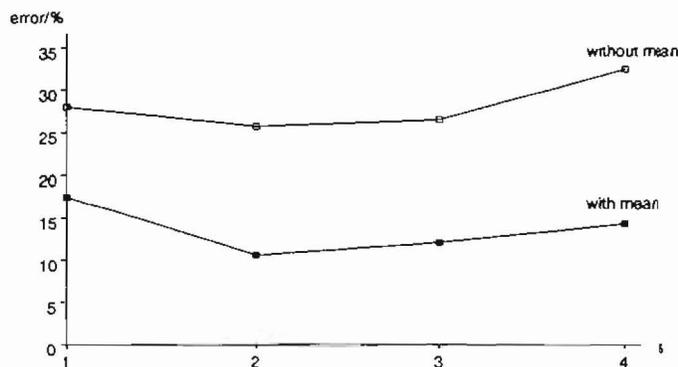


FIGURE 6.12: Accuracy of Gabor-Markov orthogonal-normal orientation classifier. Error rate against sampling period s (pixel units) for the fifth-order All-Quarters uncorrupted orthogonal-normal orientation classifier. 132 Brodatz image quarters were classified as described in §6.4.3. Error rates are higher than for the pixel or composite-feature classifiers (Figures 6.10 and 6.13).

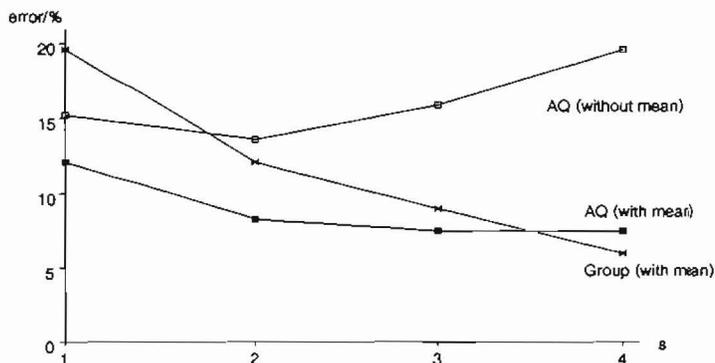


FIGURE 6.13: Accuracy of Gabor-Markov Resultant composite-feature classifier. Error rate against sampling period s (pixel units) for the fifth-order uncorrupted Resultant composite-feature classifier. 132 Brodatz image quarters were classified as described in §6.4.3, using All-Quarters (AQ) and Group methods. When the field mean is used, accuracy is similar to the unsampled pixel classifier (Figure 6.10).

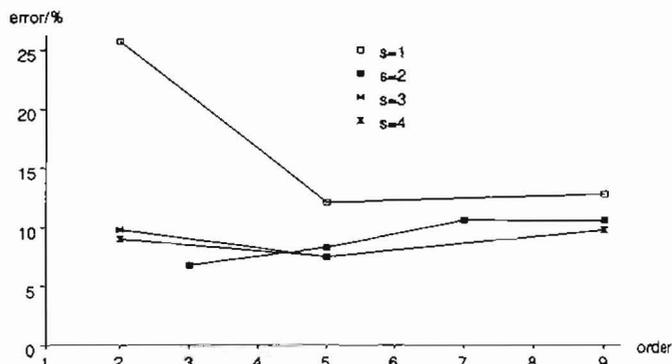


FIGURE 6.14: Accuracy of Gabor-Markov Resultant composite-feature classifier. Error rate against MRF order for the All-Quarters uncorrupted Resultant composite-feature classifier. 132 Brodatz image quarters were classified as described in §6.4.3, using field mean. Sampling period s is measured in pixel units.

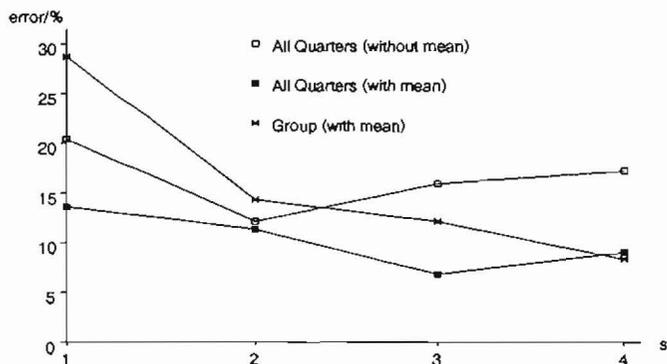


FIGURE 6.15: Accuracy of Gabor-Markov Profile composite-feature classifier. Error rate against sampling period s (pixel units) for the fifth-order uncorrupted Profile composite-feature classifier. 132 Brodatz image quarters were classified as described in §6.4.3, using All-Quarters and Group methods. When the field mean is used, accuracy is similar to the unsampled pixel classifier (Figure 6.10).

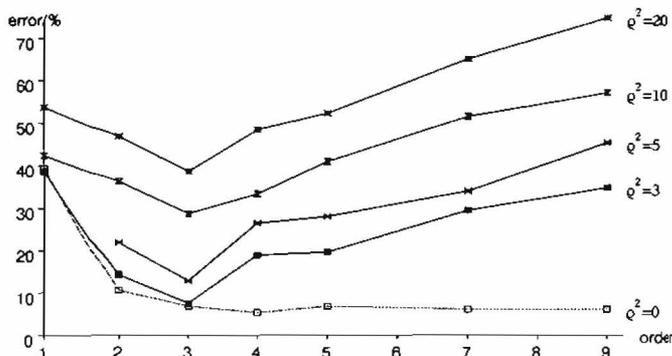


FIGURE 6.16: Accuracy of unsampled pixel classifier with noise. Error rate against MRF order for the unsampled All-Quarters true-likelihood pixel classifier with additive noise. 132 Brodatz image quarters were classified as described in §6.4.3, ignoring field mean. No noise is shown dotted; increased noise variance ρ^2 (gray-level units squared) corresponds to increasing error. Similar trends were found using the field mean or Group classifier.

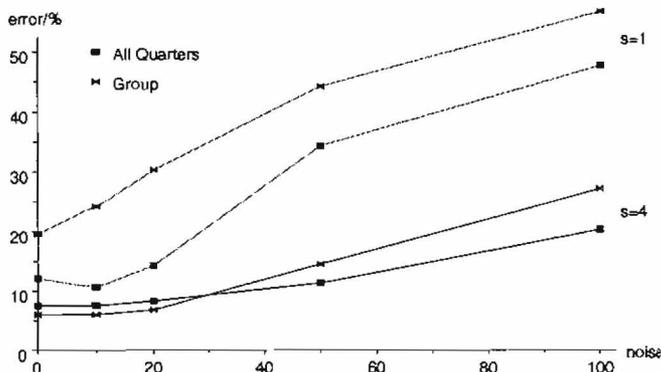


FIGURE 6.17: Accuracy of Gabor-Markov Resultant composite-feature classifier with noise. Error rate against noise variance (gray-level units squared) for the fifth-order Resultant composite-feature classifier. 132 Brodatz image quarters were classified as described in §6.4.3, using field mean, by All-Quarters and Group methods. Accuracy is higher for larger sampling period s (pixel units) but reduced when the field mean is ignored. With a sampling period of four pixels, the feature classifier is approximately 45 times less sensitive to noise than the unsampled pixel classifier (Figure 6.16).

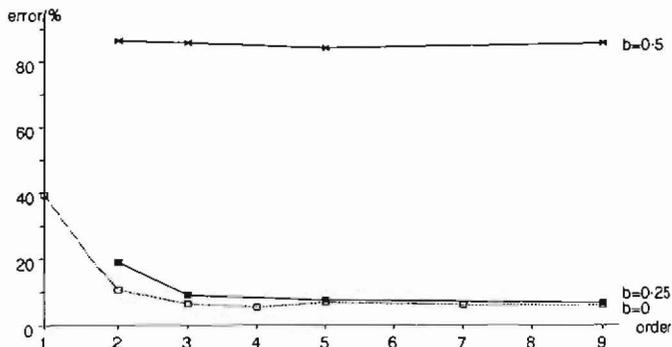


FIGURE 6.18: Accuracy of unsampled pixel classifier with blur. Error rate against MRF order for the unsampled All-Quarters true-likelihood pixel classifier with blur. 132 Brodatz image quarters were classified as described in §6.4.3, ignoring field mean. No blur is shown dotted; increasing blur parameter b (pixel units) corresponds to rapidly-increasing error. Similar trends were found using the field mean or Group classifier.

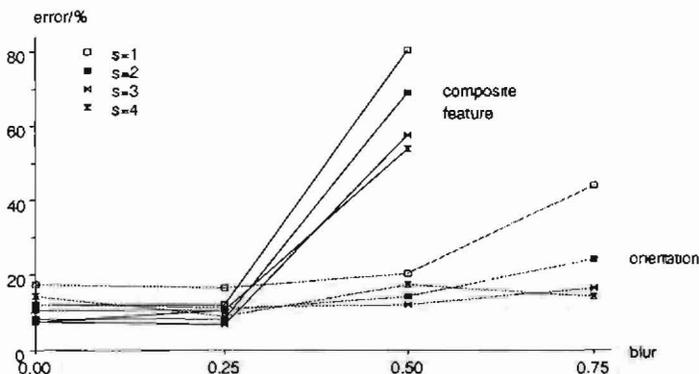


FIGURE 6.19: Accuracy of Gabor-Markov feature classifiers with blur. Error rate against blur parameter (pixel units) for fifth-order classifiers. 132 Brodatz image quarters were classified as described in §6.4.3, using field mean. **Solid:** Resultant composite-feature classifier. **Dotted:** orthogonal-normal orientation classifier. Accuracy is reduced when the field mean is ignored. Blur is less destructive for a large sampling period s (pixel units). Both feature classifiers are less sensitive to blur than the unsampled pixel classifier (Figure 6.18). The orientation classifier is particularly resilient to blur but does less well with noisy or uncorrupted images.

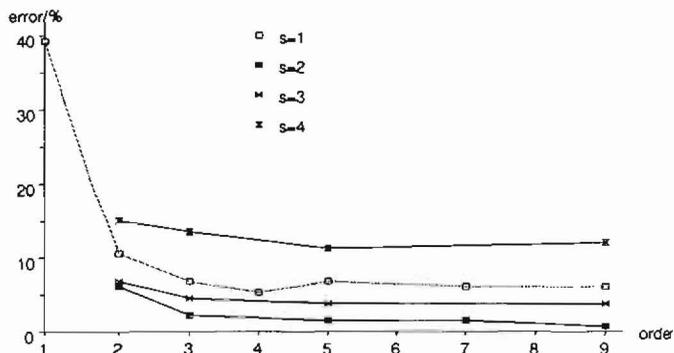


FIGURE 6.20: Accuracy of Sub-Sampled Markov classifier. Error rate against MRF order for the sampled All-Quarters true-likelihood pixel classifier. 132 Brodatz image quarters were classified as described in §6.4.3, ignoring field mean, for several sampling periods s (pixel units). No sampling ($s = 1$) is shown dotted; moderate sampling increases accuracy. Similar trends were found using the field mean or Group classifier.

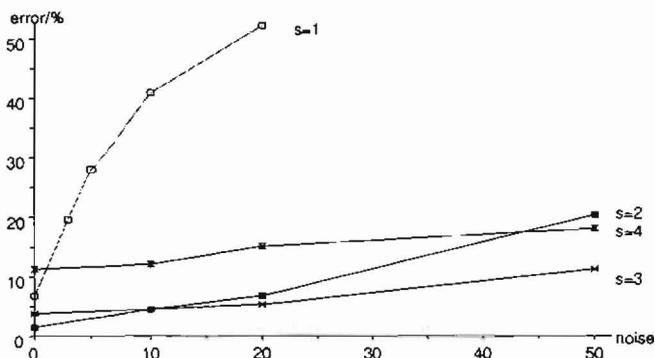


FIGURE 6.21: Accuracy of Sub-Sampled Markov classifier with noise. Error rate against noise variance (gray-level units squared) for the sampled fifth-order All-Quarters true-likelihood pixel classifier. 132 Brodatz image quarters were classified as described in §6.4.3, ignoring field mean, for several sampling periods s (pixel units). No sampling ($s = 1$) is shown dotted; moderate sampling greatly increases accuracy, slope decreasing as the sampling period is increased. Similar trends were found using the field mean, Group classifier or different MRF order.

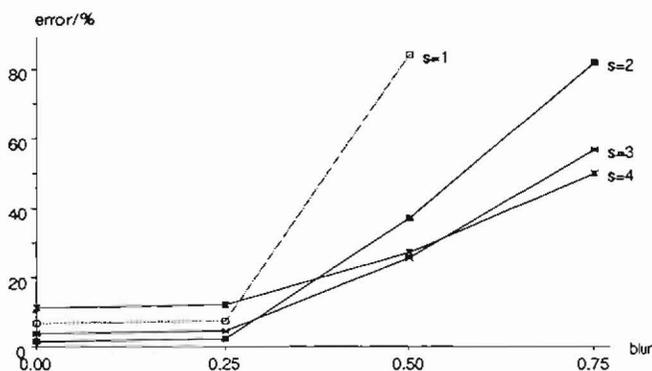


FIGURE 6.22: Accuracy of Sub-Sampled Markov classifier with blur. Error rate against blur parameter (pixel units) for the sampled fifth-order All-Quarters true-likelihood pixel classifier. 132 Brodatz image quarters were classified as described in §6.4.3, ignoring field mean, for several sampling periods s (pixel units). No sampling ($s = 1$) is shown dotted; moderate sampling greatly increases blur tolerance, slope decreasing as the sampling period is increased. Similar trends were found using the field mean, Group classifier or different MRF order.

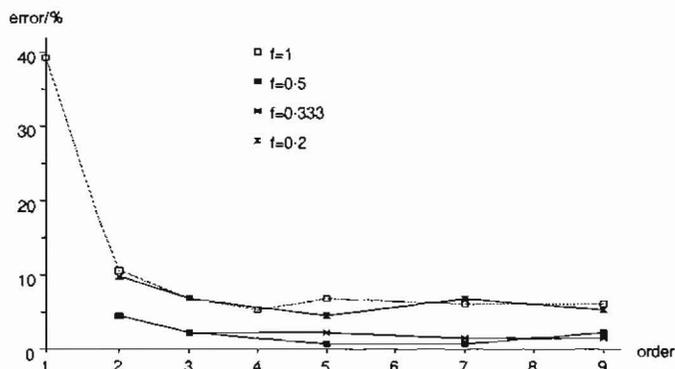


FIGURE 6.23: Accuracy of Smooth-Sampled Markov classifier. Error rate against MRF order for the smooth-sampled All-Quarters true likelihood pixel classifier. 132 Brodatz image quarters were classified as described in §6.4.3, ignoring field mean, for several sampling frequencies f (1/pixel units). No sampling ($f = 1$) is shown dotted; moderate sampling increases accuracy. Similar trends were found using the field mean or Group classifier. Smooth-sampling is more accurate than sub-sampling of the same degree (Figure 6.20).

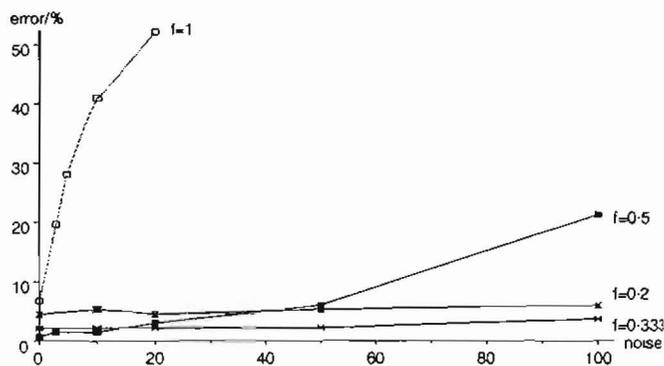


FIGURE 6.24: Accuracy of Smooth-Sampled Markov classifier with noise. Error rate against noise variance (gray-level units squared) for the smooth-sampled fifth-order All-Quarters true-likelihood pixel classifier. 132 Brodatz image quarters were classified as described in §6.4.3, ignoring field mean, for several sampling frequencies f (1/pixel units). No sampling ($f = 1$) is shown dotted; moderate sampling greatly increases accuracy, slope decreasing for lower sampling frequency. With a sampling period of five pixels ($f = 0.2$), slope is reduced relative to the unsampled pixel classifier by a factor 200 (Figure 6.16). Similar trends were found using the field mean, Group classifier or different MRF order.

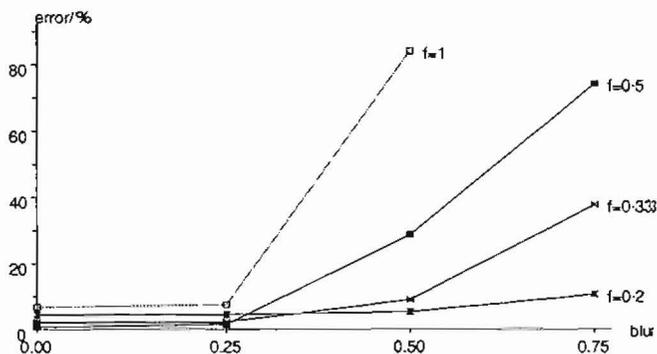


FIGURE 6.25: Accuracy of Smooth-Sampled Markov classifier with blur. Error rate against blur parameter (pixel units) for the smooth-sampled fifth-order All-Quarters true-likelihood pixel classifier. 132 Brodatz image quarters were classified as described in §6.4.3, ignoring field mean, for several sampling frequencies f (1/pixel units). No sampling ($f = 1$) is shown dotted; moderate sampling greatly increases accuracy, slope decreasing as the sampling period is increased. Similar trends were found using the field mean, Group classifier or different MRF order.

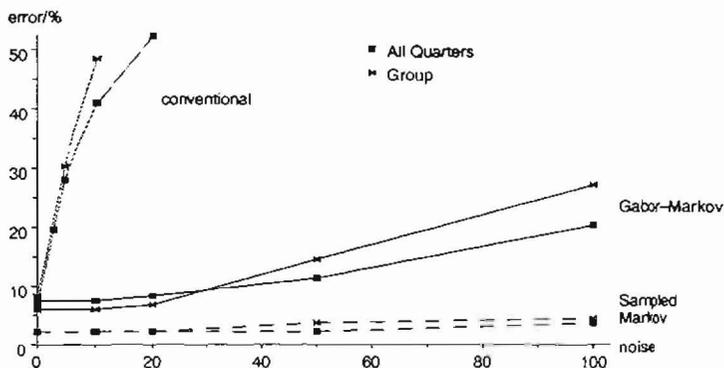


FIGURE 6.26: Comparative accuracy of Gabor-Markov, Sampled-Markov and conventional classifiers with noise. Error rate against noise variance (gray-level units squared) for fifth-order classifiers. *Dotted*: conventional classifier without field mean. *Solid*: Resultant composite-feature classifier with field mean, sampling period 4 pixels. *Dashed*: Smooth-Sampled classifier without field mean, sampling period 3 pixels. 132 Brodatz image quarters were classified as described in §6.4.3, using All-Quarters and Group methods. The Gabor-Markov classifier achieves similar basic accuracy to the conventional classifier and is much more noise-tolerant, but the Smooth-Sampled Markov classifier achieves both higher accuracy and greater noise-tolerance than Gabor-Markov.

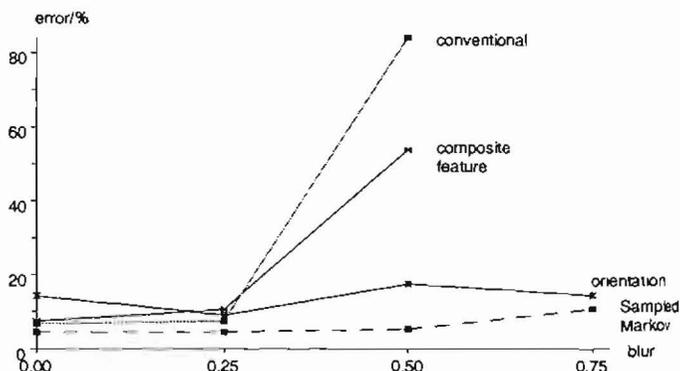


FIGURE 6.27: Comparative accuracy of Gabor-Markov, Sampled-Markov and conventional classifiers with blur. Error rate against blur parameter (pixel units) for fifth-order All-Quarters classifiers. *Dotted*: conventional classifier without field mean. *Solid*: Resultant composite-feature and orthogonal-normal orientation classifiers with field means, sampling period 4 pixels. *Dashed*: Smooth-Sampled Markov classifier without field mean, sampling period 5 pixels. 132 Brodatz image quarters were classified as described in §6.4.3. The composite-feature classifier achieves similar basic accuracy to the conventional classifier but is more blur-tolerant; greater blur-tolerance is displayed by the orientation classifier but this has a higher initial error rate. The Smooth-Sampled Markov classifier offers both higher accuracy and greater noise-tolerance than either Gabor-Markov method.

7

Consequences of Adopting Auto-Normal Models

The merits of our proposed Gabor–Markov and Sampled–Markov models were compared with those of a conventional Markovian analysis in Chapter 6 by adopting texture classification accuracy as a benchmark. Given the enhanced descriptive power available to hierarchical models (Chapter 3), and the attractive theoretical properties of Gabor filters (Chapter 4), we anticipated that our hybrid Gabor–Markov models would offer superior performance and computational efficiency. Evidence for both trends was presented in Chapter 6: Gabor–Markov models are up to 45 times more robust than the conventional approach, and may be sampled heavily without loss of accuracy.

Our appraisal of the Gabor–Markov framework led to the discovery of Sampled–Markov models possessing even more favourable attributes. These new representations are obtained simply by sampling the test image, and offer greater robustness and efficiency than Gabor–Markov models, achieving 100% classification accuracy in some circumstances. Our results have important practical consequences because there appears to be little reason to retain conventional unsampled pixel models except in special cases.

In this Chapter, we investigate possible causes for the success of Sampled–Markov and Gabor–Markov models, commenting on the influence of the imposed auto-normal structure. The level of correspondence between model and data correlates closely with classifier performance: unsampled pixel models show moderate agreement, which deteriorates rapidly when the image is corrupted; but Sampled–Markov models describe the experimental data closely. Sampled Gabor–Markov models are intermediate between these two extremes, but we suggest improvements which bring them into closer harmony with the experimental data. These observations establish the importance of

pre-processing image data prior to statistical analysis. Examination of the interaction between image corruption and the auto-normal model allows us to predict trends in misclassification error, and we demonstrate that this behaviour is consistent with our classification results. By investigating the effect of sampling on the image power spectrum, we suggest origins for the superior performance of Sampled-Markov models, concluding that it stems from close agreement between model and data.

7.1 Relation between Classifier Accuracy and Fidelity of our Statistical Model

Markov random field models used by all the classifiers studied in Chapter 6, representing both pixel and Gabor-Markov data, were assumed to be auto-normal (§5.1.5). Several approximations are known to be implicit in this choice, for example pixel data are discrete and bounded while the normal distribution is continuous and unbounded, but it is almost compulsory to adopt the auto-normal form for computational reasons (§6.1). Attempting to explain the remarkably good performance of our sampled classifiers (§6.6), we suggested on page 158 that image sampling might lead to a closer fit with the auto-normal model, thereby raising classifier efficiency. In this section, we test this hypothesis by investigating the level of agreement between our imposed statistical model and the empirical distributions of measured data, and assess the extent to which this correlates with classifier accuracy and robustness. Having identified parameters which influence the degree of correspondence, we propose positive measures to increase classification accuracy in §§7.2-7.4.

7.1.1 Modelling Error for Unsampled Pixel Images

Pixel images are commonly assumed to be multivariate normal, and have been modelled successfully by auto-normal MRFs (§6.1). Only if the fit to the normal distribution is very poor will a more cumbersome model be of greater practical use. In order to assess the acceptability of the fit, we examined the local conditional error distribution, $f(u) = f(x - \hat{x})$, where, for each image site, x represents the gray-level pixel value, \hat{x} the value predicted from the MRF neighbourhood structure, and u (gray-level units) is the MRF predictor-error (5.27 on page 108). For a true auto-normal distribution, we expect the predictor-error to be normally-distributed, $u \sim \mathcal{G}(0, \sigma^2)$, and we shall compare the observed distributions with this form. Visual comparison is sufficient to obtain a qualitative measure of the fit and is helpful in suggesting how it might be improved. A more formal goodness-of-fit analysis is only able to confirm or deny the identity hypothesis (e.g. Besag, 1974). Note that the MRF predictor-error u has zero mean (§5.1.2), and that by “larger predictor-error” we mean a larger *variance*, σ_u^2 .

We investigated the MRF predictor-error distribution for models estimated from our Brodatz set (Figure 1.6 on page 11) and some additional images used in Chapter 4, taking several values for MRF order and using both true and pseudo likelihood. Typically, the general form of the observed MRF predictor-error distribution is correct, but

agreement with the normal curve is far from exact, irrespective of whether the Brodatz texture appears homogeneous. Error distributions $f(u)$ for two representative images are shown in Figure 7.1; histogram bin size was adjusted to avoid excessive noise but there are sufficient pixels in these images for it to appear continuous. The MRF model variance chosen by the pseudo-likelihood estimation method is necessarily the same as the empirical predictor-error variance s_u^2 (§6.2.1), and can mean that the pseudo-likelihood histogram appears to follow the expected form more closely than for true-likelihood estimation. In the latter case, the likelihood is weighted according to the joint probability distribution (6-4 on page 139), and hence the model and experimental predictor-error variances may differ. It has been suggested that typical spatial variation of gray-level images is described more aptly by the Laplacian rather than the normal distribution, but we did not pursue this possibility (Trussell & Kruger, 1978).

By comparison with the poor agreement obtained for real images, the fit to a synthetic image, produced by the Fourier transform method (§5.3.5), is excellent (Figure 7.2a). When the synthetic image is expressed in gray-level units rather than in floating point, agreement is a little less exact, with a trend to larger absolute error and noticeable “tails” (data not shown), but is still far superior to that of real images (Figure 7.1). Despite this, the auto-normal model usually describes real images sufficiently well to distinguish mis-matches (Figure 7.2b).

Possible Causes of Modelling Error

Imperfect agreement of the type noted above for real textures occurs with the exact image used for parameter estimation, and hence represents modelling error rather than a lack of robustness. Several factors may contribute to the poor fit, including:

- poor choice of MRF distribution function (data not auto-normal);
- failure of the parameter-estimation method;
- wrong order for MRF;
- error introduced by toroidal boundary approximation;
- quantisation error;
- clipping to pixel units;
- “noise”;
- image not homogeneous;
- image or camera blur.

All of these probably have some effect. The imperfect fit for unsampled pixel data (Figure 7.1) could certainly be blamed on the wrong choice of MRF model, but this is really only a re-description of the result. More interesting questions are why classifier performance is improved by sampling, and how modelling error may be reduced or overcome. Modelling error represents a failure of the image representation because

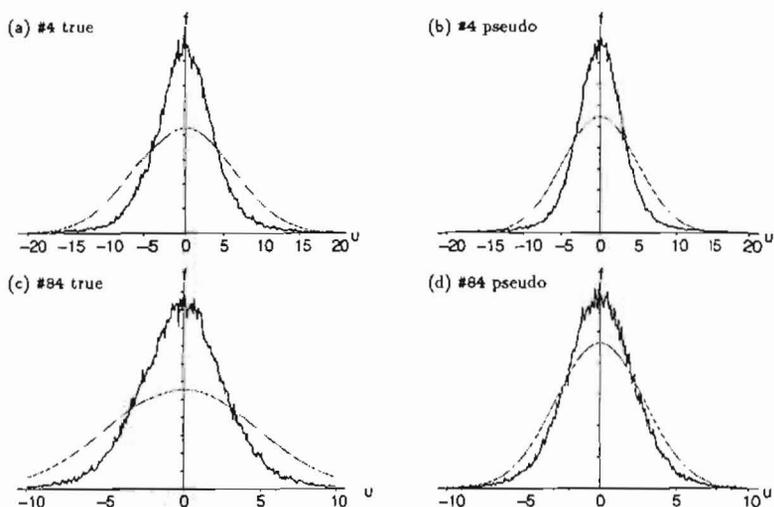


FIGURE 7.1: Agreement with auto-normal model of unsampled Brodatz images. Histograms of observed (solid) and expected (dotted) distributions for ninth-order MRF predictor-error u (gray-level units), to different vertical scales, using the analysis described in §6.2. (a) pressed-cork image (#4; Figure 1.6 on page 11), true-likelihood parameters; (b) #4, pseudo-likelihood parameters. (c) raffia image (#84; Figure 1.6), true-likelihood parameters; (d) #84, pseudo-likelihood parameters. These results are typical: in no case is agreement close despite apparent homogeneity of the original textures.

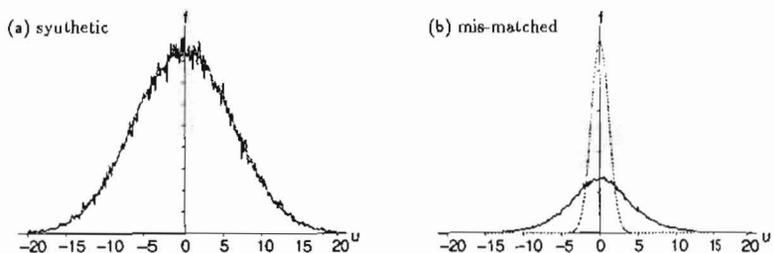


FIGURE 7.2: Agreement with auto-normal model of synthetic and mis-matched textures. Histograms of observed (solid) and expected (dotted) distributions for ninth-order true-likelihood MRF predictor-error u (gray-level units), to different vertical scales, using the analysis described in §6.2. (a) synthetic pressed-cork image (#4; Figure 6.2 on page 142) evaluated with correct parameter set; (b) genuine expanded mica image (#5; Figure 1.6 on page 11) evaluated with parameter set for #4.

an image should be definitive once it has been recorded, irrespective of the physical processes giving rise to the observed texture.

In order to assess the relative influence of each potential source of error in isolation, we took a *synthetic* image, which initially displays perfect agreement (Figure 7.2), modified it appropriately, re-estimated new MRF parameters, and examined the resulting fit. This procedure ensures that only modelling error is introduced because the descriptive accuracy of each MRF parameter set is evaluated with the exact image from which it was derived. The ninth-order true-likelihood synthetic image used for this purpose is shown in Figure 6.2 on page 142; its Brodatz parent #4 “pressed cork” in Figure 1.6 on page 11. True-likelihood manipulation was used throughout.

Failure of the Parameter-Estimation Method

True-likelihood parameter estimation is not straightforward because the likelihood function is multi-dimensional and may have many local maxima. No analytical estimate is possible for the optimal parameter set, and there is a clear danger that the gradient-ascent algorithm we employ (page 139) will become trapped in a local maximum. Having a large image may exacerbate this problem because there is potential for the likelihood function to be even more uneven, and improved performance observed from the Sampled-Markov classifiers is consistent with this possibility (§6.6).

We argued in §6.2.1 (page 139) that failure of the parameter-estimation algorithm does not introduce significant error in practice, having found that repeated gradient ascent from different initial points was unable to improve the parameter estimate. This contention is further strengthened because advantages of sampling are also observed for pseudo-likelihood parameters, which are not obtained by gradient ascent. In order to confirm this conclusion, we compared the empirical fit to an entire Brodatz image with that obtained from a single quarter. If image size does interact with parameter estimation in the way we hypothesised above, agreement with the quarter-image should be superior to the whole, but no significant improvement was found (data not shown), consistent with our belief that this is indeed a negligible source of error.

Incorrect MRF Neighbourhood Size

When the neighbourhood size of the Markov random field model is increased, the new parameter space contains the old, and hence it is always possible to obtain agreement at least as good as before. When the Markovianity of the data is only approximate, increasing the MRF neighbourhood size always leads to a better description of the data. Closer agreement is demonstrated by the MRF predictor being more accurate, leading to a lower variance, and by the predictor-error distribution being approximately normal. As expected, the average MRF variance of true-likelihood parameter sets estimated from all our 132 Brodatz image segments is a decreasing function of MRF order (Figure 7.3). For unsampled images, the variance decreases by a factor 5-7 between third and ninth order.

The importance of having the correct order for the model is demonstrated in a more controlled way by modelling a ninth-order synthetic image with a reduced parameter

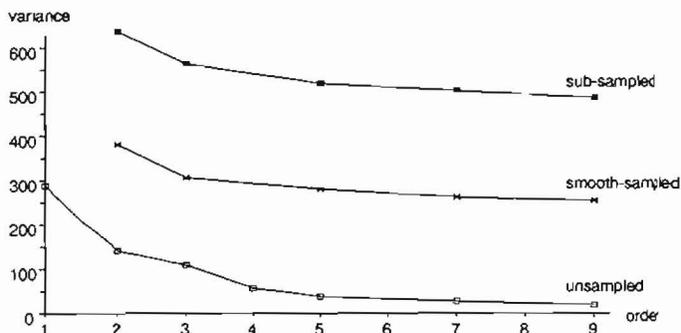


FIGURE 7.3: **Influence of sampling period and MRF order on variance of pixel models.** Average MRF variance (gray-level units squared) from true-likelihood parameter sets estimated from our 132 Brodatz image quartets, against MRF order. Sampling was done with a period of 2 pixels. In each case, model variance is a decreasing function of MRF order.

set (Figure 7.4a, b). Note that because of the inter-dependence of MRF coefficients, the new parameter vectors are not simply truncated versions of the old. The degree of error caused by simplifying the MRF neighbourhood structure clearly depends on the nature of the original Markov random field, but significant error occurs in this case, which is typical, caused by the inability of the reduced model to follow the shape of the image power spectrum closely. Agreement with the seventh-order model (Figure 7.4b) is closer than for fifth order (a), and when MRF neighbourhood size is increased to the correct value, correspondence is almost exact (Figure 7.2). This transition is more gradual for real images (data not shown), but our observations suggest that an increase in MRF neighbourhood size is always beneficial. In practice, however, increasing MRF order beyond third-order had very little impact on classifier performance, and larger order led to *less* accurate classification in some cases (§6.5). Most sampled models of low MRF order achieve good classification and modeling accuracy (§6.6; §7.1.3), and although error rates are high for *very* small neighbourhood size, it does not appear that the order of the MRF is generally of paramount importance.

Toroidal Boundary Conditions

It appears unlikely that the toroidal boundary approximation is a significant source of error because the image dimensions are much larger than the extent of the MRF neighbourhood. This was confirmed by comparing the fit under toroidal and free boundary conditions for real and synthetic textures. As expected, there was little difference between them (data not shown).

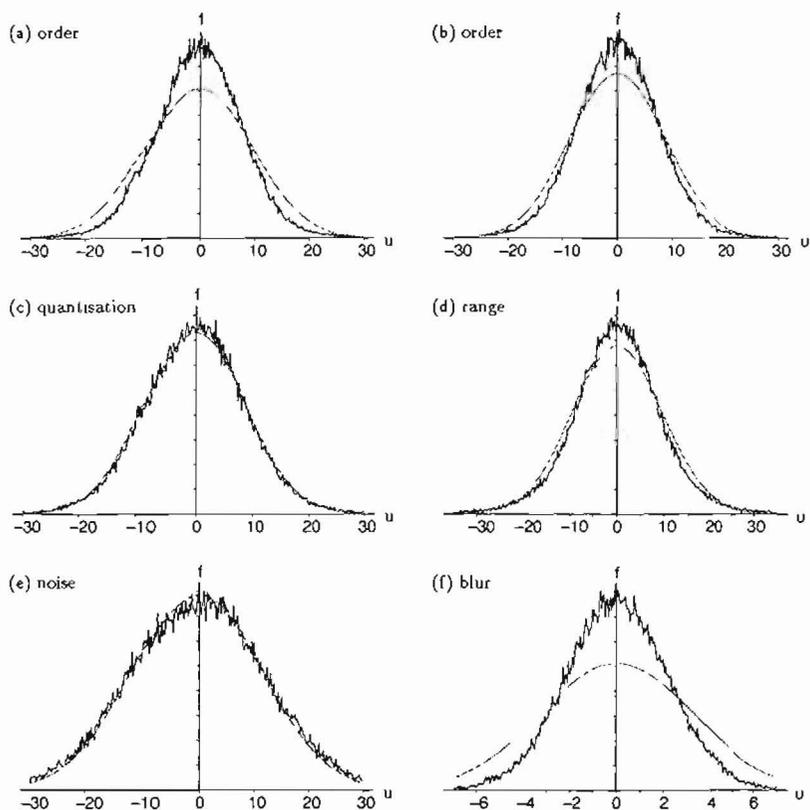


FIGURE 7.4: Agreement with auto-normal model of perturbed synthetic textures. Histograms of observed (solid) and expected (dotted) distributions for MRF predictor-error u (gray-level units), using the analysis described in §6.2, and a 256×256 synthetic pressed-cork image (#4). (a) ninth-order image (Figure 6.2 on page 142) and fifth-order parameter set; (b) ninth-order image and seventh-order parameter set; (c) ninth-order image quantised to 5 bits (32 gray levels) and ninth-order parameter set; (d) ninth-order image clipped to gray-level range [32, 223] and ninth-order parameter set; (e) fifth-order image (Figure 1.4a on page 7) with added noise of variance 20 gray-level units squared, and fifth-order parameter set; (f) fifth-order image blurred with parameter 0.5 pixels, and fifth-order parameter set. Errors are introduced when the model has insufficient parameters, by clipping, and by blur, but moderate levels of quantisation or Gaussian noise have little effect.

Quantisation Error

Pixel images are normally represented using 8 bits, giving 256 (unsigned) gray levels. This is a potential source of error because image data is assumed to be continuous by the auto-normal model. In order to test whether quantisation error has a significant impact, we re-estimated parameters for the ninth-order true-likelihood synthetic pressed-cork image (#4; Figure 6.2) quantised to only 5 bits (32 gray levels). Agreement between the observed MRF predictor-error distribution and the normal curve is very good, suggesting that quantisation error is, at most, a very minor effect (Figure 7.4c). Note that this fit was for MRF parameters re-estimated from the quantised image rather than the original set; correspondence with the latter was much poorer (data not shown).

*Clipping Error*¹

The range of the normal distribution is unbounded, whereas pixel images are bounded above and below. Imaging parameters were adjusted to make good apparent use of the available dynamic range when our textures were photographed, but this was not checked quantitatively. It is possible that clipping at one extreme could cause skew within the distribution, and clipping certainly distorts the Markov property by disrupting the correlation structure.

In order to test these effects, the synthetic pressed-cork image was clipped artificially, its MRF parameters re-estimated, and the new MRF predictor-error distribution compared to the normal curve (Figure 7.4d). Modelling error is modest, but is larger if the order of the model is decreased, and it does not appear to matter whether or not clipping is symmetrical (data not shown). Agreement may be improved a little by describing the range-restricted synthetic image by a model of higher MRF order than the original. The extent to which an image is clipped is determined by the camera and post-processing parameters, which may be controllable. Our results show that clipping does interfere with texture modelling, perhaps to a greater extent than quantisation effects, suggesting that the image acquisition environment should be adjusted carefully.

Additive Noise Error

The noise parameters of our imaging system are unknown but it is likely that noise depends on the image data whereas our simple model assumes it to be white, Gaussian, additive and independent of the image (§6.4). It could be argued that once the image has been taken, any noise should be regarded as an intrinsic part of the texture rather than due to some external influence.

In practice, small amounts of additive white noise, applied according to the classification procedure given in §6.4.3, appear not to cause the model any difficulty (Figure 7.4e). Our original texture in this case was a fifth-order synthetic pressed-cork image (Figure 1.4a on page 7), with model and pixel variances of 80 and 2750 gray-level units squared, respectively. The fit shown is after addition of noise of variance 20

¹Clipping occurs when a signal exceeds the dynamic range of the measuring device and is recorded at the nearest available value.

— this is sufficient to cause a nine-fold increase in errors with the dense pixel classifier when using genuine Brodatz textures (§6.5.2).

Inhomogeneous Texture Samples

Large inhomogeneities in the texture will certainly prevent it being described accurately by a homogeneous model. However, no significant difference was observed between agreement obtained with apparently-homogeneous and apparently inhomogeneous Brodatz images (data not shown). Although none of our images is perfectly homogeneous, the lack of any obvious relationship does suggest that the primary cause lies elsewhere.

It has been suggested that non-stationary or “coloured” Gaussian MRFs may provide a closer description of real images than the form we have used (§6.1), allowing the field mean to vary with image location (Hunt & Cannon, 1976; Silverman & Cooper, 1988). When image mean varies smoothly with position, it may be estimated by a local weighted average, and hence a zero-mean field recovered by convolving the image with a blur kernel and subtracting it from the original (Hunt & Cannon, 1976). We use this procedure to eliminate the local field mean during Gabor filtering (§A.24). When our Brodatz image set was processed in this fashion and the classification exercise repeated with a conventional ninth-order true-likelihood classifier, the error rate was hardly affected (6.8% against 6.1% previously), confirming that our textures are largely homogeneous with respect to field mean.

Image Blur

Several types of blur are possible, including optical blur and camera leakage, but blur may also be a valid property of the texture (water ripples, for example). The actual nature of the “blur” transformation is complex, but we adopt uniform Gaussian blur as a simple model (§6.4). As with noise, once a photograph has been taken, the blur may be considered to be an intrinsic part of the resulting image.

We assessed the importance of image blur by comparing the MRF predictor-error distributions obtained from blurred synthetic images with the normal curve (Figure 7.4f), and found that the new MRF models were unable to describe the structure of the blurred images adequately, failing in a very similar way to the mis-fit seen for Brodatz originals (Figure 7.1). It appears that a blurred image is no longer well-described by an auto-normal MRF. Visually, the blurred texture is still quite recognisable (Figure 6.5 on page 154), and so errors appear to be caused by a failure of the model, for which possible causes are discussed in §7.2.1.

Summary of the Causes of Modelling Error

Of the effects listed above, blur introduces the largest modelling errors, followed by clipping; both are difficult to eliminate entirely from real images. The degree of blur required to introduce significant modelling error (and also a significant increase in classification error) was small, space-constant $b = 0.5$ pixels. Unless great care is taken, or computationally-expensive restoration employed, blur is likely to represent a significant

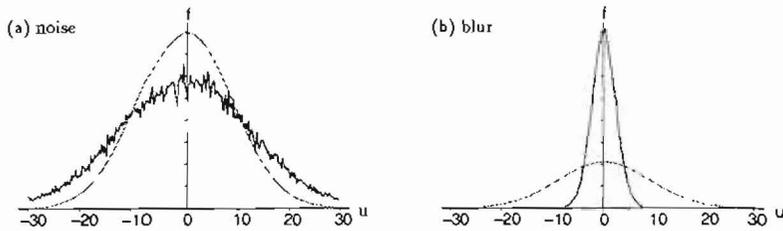


FIGURE 7.5: Agreement with auto-normal model of corrupted synthetic images. Histograms of observed (solid) and expected (dotted) distributions for fifth-order MRF predictor-error u (gray-level units), to different vertical scales, using the analysis described in §6.2. Agreement with the uncorrupted fifth-order 256×256 synthetic pressed-cork image (#4; Figure 1.4a) is excellent; we corrupted this image before matching it against the original parameter set. (a) after addition of noise, variance 20 gray-level units squared; (b) after blur, parameter 0.5 pixels.

source of error for real images, and we suggest in §7.2.3 how its destructive influence can be diminished. Distortion was also observed when the model had insufficient parameters, but our experience with the classifier suggests that increasing the order of the model beyond a reasonable minimum level often fails to improve accuracy (§6.5.1).

7.1.2 Robustness of Unsampled Pixel Models

Two sources of error are present in the classifier results: failure to model the original data closely, and distortion caused by image corruption (§6.5). Only the former was investigated above because fresh parameter sets were re-estimated for each perturbed image. In this section, we examine the impact of image corruption on the level of agreement obtained with the original model.

In order to isolate the effects of modelling and perturbation error, we again used a synthetic texture which initially agreed with the model precisely, choosing a fifth-order copy of the pressed-cork image (#4; Figure 1.4a). Addition of noise of variance 20 gray-level units squared introduced significant disagreement between the normal and observed distributions (Figure 7.5a): MRF predictor-errors were higher than anticipated. This is consistent with our intuition that noise reduces the accuracy of the MRF predictor. Noise of this magnitude increased the error rate of the fifth-order classifier from 6.8% to 52.2% when using genuine Brodatz textures (§6.5.2).

Blur has an even more powerful effect (Figure 7.5b): MRF predictor-errors are tightly bunched around zero, confirming that blur reduces sudden image variation. The distribution shown was for a moderate amount of blur, parameter $b = 0.5$ pixel, sufficient to increase the error rate of the fifth-order classifier from 6.8% to 84% (§6.5.3). Such low accuracy is not hard to understand given the high level of disagreement seen here.

All images show a similar pattern but the rate of degradation varies, and the noise-tolerance of the pressed-cork image is a little higher than average. It is important to

distinguish these results from those of the previous section: modelling error relates primarily to the level of classification error for uncorrupted images; whereas the rate at which the model degrades when the image is corrupted determines classifier robustness. Whilst a model which is initially in good agreement with the data may also lead to a more robust classifier, the two effects need not vary in a similar way.

7.1.3 Descriptive Accuracy of Sampled-Markov Models

We have shown above that dense pixel models do not fully capture image structure (§7.1.1), and are sensitive to small image perturbations (§7.1.2). These deficiencies are reflected in classifier performance as a high error baseline and poor robustness (§6.5). The puzzle is how these problems are overcome by sampling, which leads to both higher accuracy and greater robustness (§6.6).

Modelling Error

As with unsampled models (§7.1.1), we shall investigate modelling error by evaluating MRF parameter sets in conjunction with the exact image from which they were estimated. Histograms of ninth-order true-likelihood MRF predictor-error distributions $f(u)$ for sub-sampled Brodatz images are shown in Figure 7.6 for the same textures examined previously with unsampled models (compare with Figure 7.1), #4 pressed-cork and #84 raffia (Figure 1.6 on page 11). For both images, sampling has transformed the level of agreement from poor to very good, despite the residual effect of the factors listed on page 177. Even closer agreement is obtained by the corresponding Smooth-Sampled models, and modelling error is virtually eliminated (Figure 7.7). This was confirmed by looking at the MRF predictor-error field u , which appeared more homogeneous after sampling and had highest magnitude for the Sub-Sampled Markov models (data not shown). The MRF variance is higher for Smooth-Sampled Markov than conventional models, and for Sub-Sampled models it is higher still, although the proportional change between third and ninth order is greatest with unsampled models (Figure 7.3). To some extent, the reduced variance for a smooth-sampled image reflects a reduction in pixel variance caused by compression of the dynamic range.

Following the procedure outlined in §7.1.1, we attempted to isolate the effects of particular types of modelling error by using synthetic images. The dense image was perturbed and sampled, and a new parameter estimate obtained. For the range of parameters used previously, no significant error was introduced (data not shown), in strong contrast to unsampled models (Figure 7.5).

Model Robustness

In order to assess the robustness to image noise and blur of the close-correspondence observed between experimental and normal MRF predictor-error distributions, we adopted a similar technique to that described above (§7.1.2). A 256×256 pixel image was synthesised from an unsampled fifth-order true-likelihood MRF parameter set (Figure 1.4a), derived as previously from the pressed-cork image (#4), and fresh parameter sets were

estimated after appropriate image sampling. We corrupted the unsampled image using the same techniques as above (§6.4), and compared the new level of agreement between the original parameter sets and the corrupted image after appropriate sampling. Similar trends to those observed for dense images were found: a shift towards higher MRF predictor-error for noisy images, and towards lower predictor-error after blur, but sampling reduced the extent of both effects. Fifth-order MRF predictor-error distributions obtained from a corrupted synthetic image show only limited distortion (Figure 7.8), contrasting with the same image, noise and blur parameters used previously without sampling (Figure 7.5). Blur has a stronger effect for lower sampling period, and affects sub-sampled models more than smooth-sampled, but agreement is always superior to that obtained without sampling.

Summary: Fidelity of Sampled-Markov Models

Sampled MRF models display good agreement with real textures and are resilient in the face of image corruption. These properties are reflected in the improved accuracy and robustness of the sampled classifiers. Smooth-sampled models have a lower variance than sub-sampled, and achieve a better fit to the data in some cases.

7.1.4 Descriptive Accuracy of Gabor-Markov Models

As with conventional and Sampled-Markov models, computational concerns led us to implement Gabor-Markov classifiers using auto-normal MRFs. Three types of Gabor-Markov model were proposed in §6.3, describing arrays of Gabor features directly, or after the orthogonal fields or composite-feature transformations. By its nature as a windowed linear filter, the Gabor kernel introduces blur into its output (§4.1). Consequently the shortcomings seen with blurred pixel images (§7.1.1) are also expected in dense feature images, perhaps to a greater degree reflecting the larger space constant ($\sigma_{\text{Gabor}} = 2.25$ pixels).

Direct Normal Model

Under the direct normal model (§6.3.1), the orientation feature θ is assumed normally distributed, ignoring periodicity. Correspondence between the observed MRF predictor-error and normal distributions is extremely poor (Figure 7.9a): the trend seen above for unsampled images towards a large central peak and long tails has been exaggerated, consistent with the influence of blur (§7.1.1). Sub-sampling has a dramatic effect, eliminating the central peak, but the fit is very strained because of skew (Figure 7.10a).

One way to reflect the periodicity of the data is to apply a trigonometric transformation (§6.3.1), but taking the tangent of the measured orientation makes the fit worse (Figure 7.9b). The predicted distribution is indistinguishable from the axis because the model variance is very large, presumably because \tan is insufficiently localised around the origin. Sub-sampling improves the fit a little, but it is still very poor (data not shown).

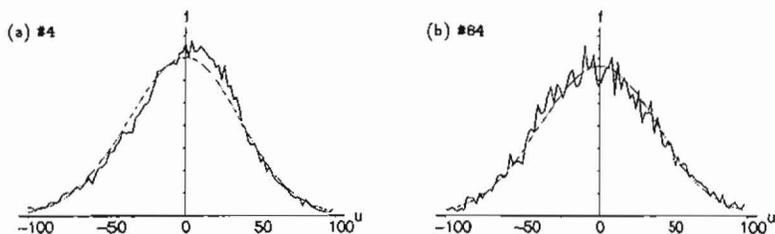


FIGURE 7.6: Agreement with auto-normal model of sub-sampled Brodatz images. Histograms of observed (solid) and expected (dotted) distributions for ninth-order MRF predictor-error u (gray-level units), to different vertical scales, using the analysis described in §6.2. (a) pressed cork image (#4; Figure 1.6 on page 11), sampling period 2 pixels; (b) raffia image (#84; Figure 1.6), sampling period 3 pixels. In both cases, agreement is far better than without sampling (compare Figure 7.1).

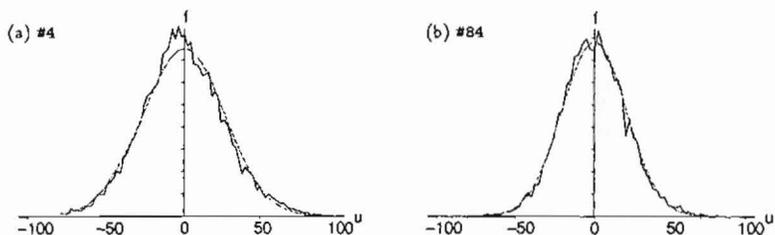


FIGURE 7.7: Agreement with auto-normal model of smooth-sampled Brodatz images. Histograms of observed (solid) and expected (dotted) distributions for ninth-order MRF predictor-error u (gray-level units), to different vertical scales, using the analysis described in §6.2. (a) pressed-cork image (#4; Figure 1.6 on page 11), sampling period 2 pixels; (b) raffia image (#84; Figure 1.6), sampling period 3 pixels. In both cases, agreement is far better than without sampling (compare Figure 7.1). Note the decreased variance compared with the corresponding sub-sampled models (Figure 7.6).

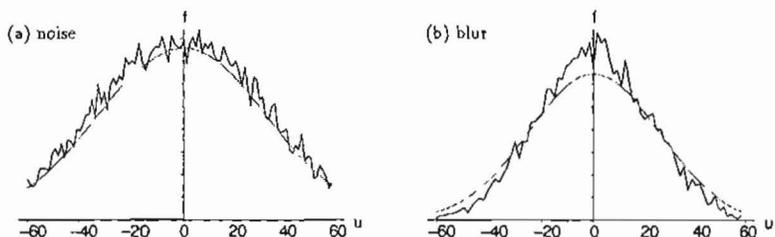


FIGURE 7.8: Agreement with auto-normal model of sampled corrupted synthetic textures. Histograms of observed (solid) and expected (dotted) distributions for fifth-order MRF predictor-error u (gray-level units), to different vertical scales, using the analysis described in §6.2. Agreement with the uncorrupted sampled fifth-order *synthetic* pressed-cork (#4; Figure 1.4a) image is excellent. We corrupt this image, sample it, and match it against the original parameter set. (a) after addition of noise, variance 20 gray-level units, and sub-sampling, period of 2 pixels; (b) after blur, parameter 0.5 pixels, and smooth-sampling, period 3 pixels. Agreement is far superior to that obtained without sampling (Figure 7.5).

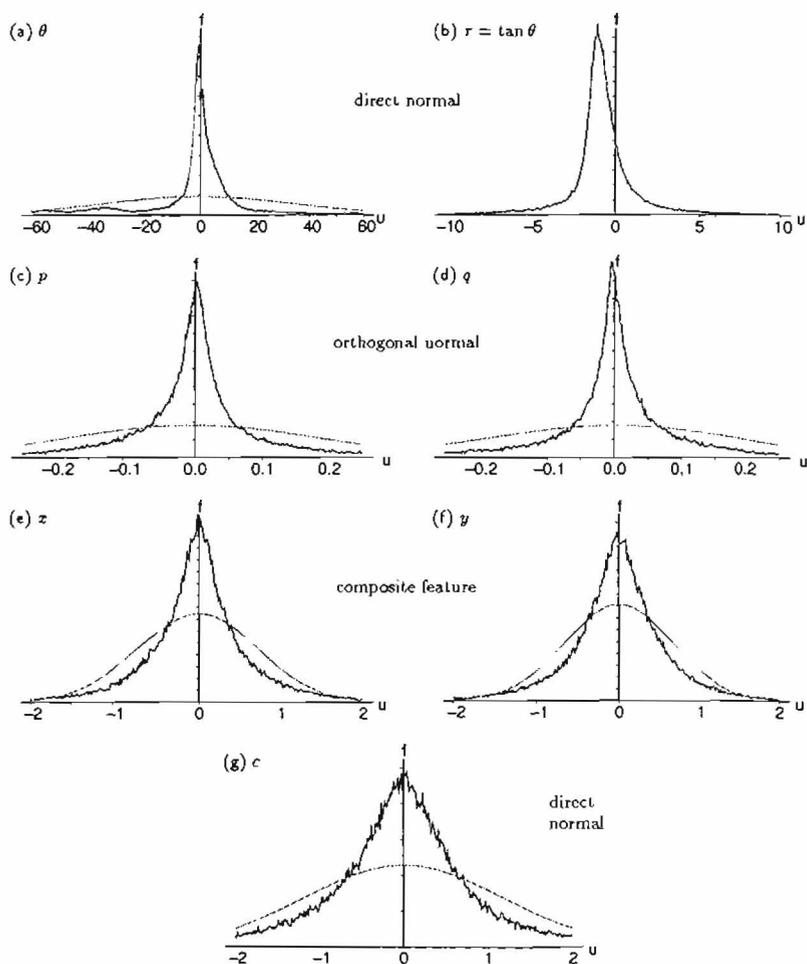


FIGURE 7.9: Agreement between auto-normal and Gabor-Markov models. Histograms of observed (solid) and expected (dotted) distributions for ninth-order MRF predictor-error u (arbitrary units), to different vertical scales, using the analysis described in §6.3. Gabor features were obtained from the pressed-cork image (#4; Figure 1.6 on page 11) by the Resultant method. (a) orientation feature θ and direct normal model; (b) $\tau = \tan \theta$ and direct normal model; (c,d) orthogonal normal fields model; (e,f) composite-feature model; (g) contrast feature c and direct normal model. In no case is agreement good.

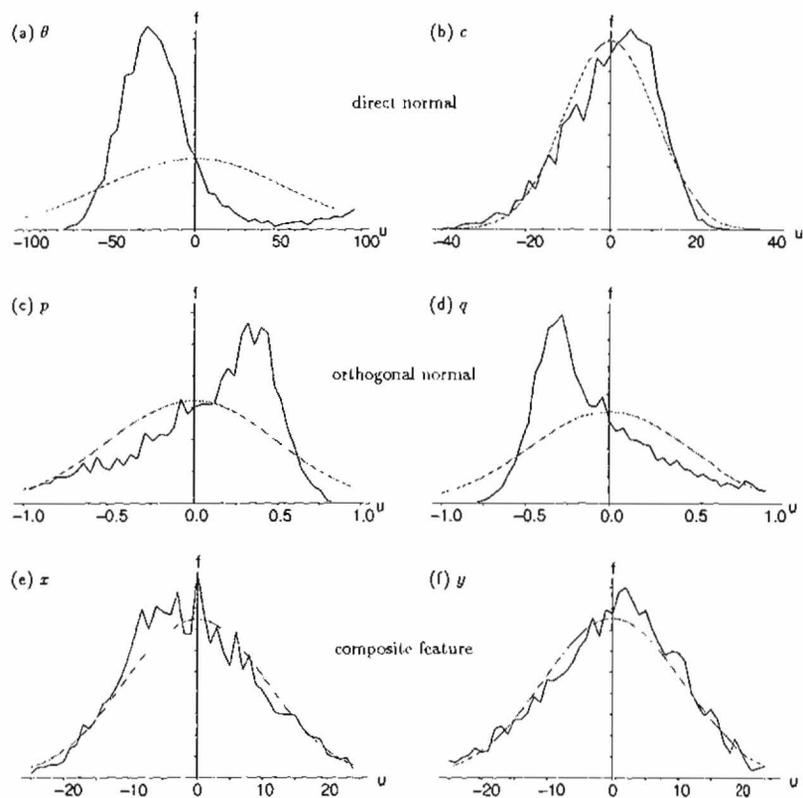


FIGURE 7.10: Agreement between auto-normal and sampled Gabor-Markov models. Histograms of observed (solid) and expected (dotted) distributions for ninth-order MRF predictor-error u (arbitrary units), to different vertical scales, using the analysis described in §6.3. Gabor features were obtained from the pressed-cork image (#4; Figure 1.6 on page 11) by the Resultant method, and sampled with a period of 4 pixels. (a) orientation feature θ and direct normal model; (b) contrast feature c and direct normal model; (c,d) orthogonal normal fields model; (e,f) composite-feature model. Agreement is better than for unsampled features (compare Figure 7.9), but (a) and (c,d) show pronounced skew.

Compared to the angular fields, the fit for the contrast feature is reasonable (Figure 7.9g), similar to that seen for pixel images (Figure 7.1). Sampling again improves the fit but also makes skew more apparent (Figure 7.10b). Resultant features were used here (§4.6), but a similar pattern is observed with the Profile LINE and SINU methods (data not shown).

Orthogonal Normal Fields Model

Our assumptions of normality and independence for the components of the orthogonal-normal model were somewhat questionable (§6.3.1), and the observed fit is again poor (Figure 7.9c,d). Note that the orthogonal normal fields model uses two Markov random fields, \mathbf{p} and \mathbf{q} , to describe the spatial variation of the data (§6.3.1). A large central peak is again prominent in unsampled fields, and when this is eliminated by sub-sampling, skew becomes apparent (Figure 7.10c,d). These results are similar to those obtained for the direct normal model. The amount of skew varies from image to image but tends to have an opposite sign for the orthogonal components. Below (§7.1.5), we suggest how the skew could be reduced in an improved model.

Composite Feature Model

There is a little more support (§A.4.2) for both the normality and independence of the components of the composite feature model (§6.3.2) compared with the orthogonal-normal model (§6.3.1). Note that two Markov random fields are employed, \mathbf{x} and \mathbf{y} . Correspondence between the unsampled experimental MRF predictor-error distributions and the normal curve is still poor, but improves upon previous results for angular fields (Figure 7.9). When Gabor features are sub-sampled, as was our original intention (§5.7), modelling accuracy is much improved, attaining an acceptable level of agreement (Figure 7.10e,f). In this respect, the composite-feature model is superior to the direct normal model, and it achieves the best fit for any of the Gabor feature data. Agreement is good for both random field components of the composite-feature model (\mathbf{x} and \mathbf{y}), with no significant skew.

We tested the robustness of the composite-feature classifier by examining the correspondence between the true parameter set and Gabor features extracted from a corrupted image. Noise of variance 20 gray-level units squared poses little difficulty, but after moderate blur (parameter 0.5 pixels), the familiar shift towards lower MRF predictor-error is observed (Figure 7.11). Both the raw fit and tolerance to corruption of the sampled composite-feature model are superior to those of dense pixel models but do not match those of sampled pixel models. This is reflected in the performance of the classifier (§6.7).

7.1.5 Improved Gabor-Markov Models

A common factor in the observed Gabor-Markov predictor-error distributions for unsampled feature data is the presence of a large central peak with long tails (Figure 7.9). The position is improved by sub-sampling but skew emerges as a new error for all except

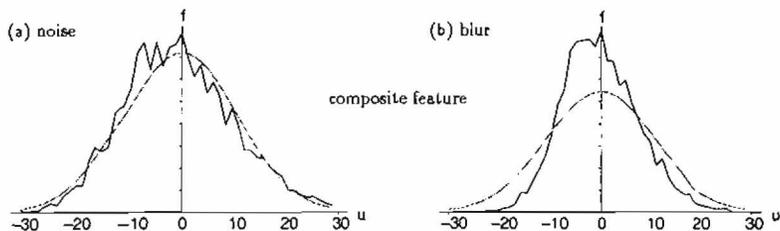


FIGURE 7.11: Agreement between auto-normal and sampled Gabor-Markov models with corruption. Histograms of observed (solid) and expected (dotted) distributions for ninth-order MRF predictor-error u (arbitrary units), to different vertical scales, using the analysis described in §6.3. Gabor features were obtained from the pressed-cork image (#4; Figure 1.6 on page 11) by the Resultant method, with a sampling period of 4 pixels, and processed using the composite-feature model. Field x is shown (y is similar), and agreement with features extracted from the uncorrupted image was good (Figure 7.10). (a) after addition of noise, variance 20 gray-level units squared; (b) after blur, parameter 0.5 pixels. Correspondence with the corrupted image is far superior to that obtained by the unsampled pixel model (Figure 7.5) but less good than that of the sampled pixel models (Figure 7.8).

the composite-feature model (Figure 7.10), because of interaction between the imposed normal model and the periodicity of angular data. Errors are introduced when adjacent values are instead treated as extremes, and the extent to which this causes skew depends on the spread of the angular data across its principal segment. This is illustrated for the orientation feature in Figure 7.12: in (a), the modal (peak) orientation occurs well to one side of the segment, leading to skew on truncation. Much less skew is introduced in (b) because the peak lies near the centre of the region and the spread is roughly symmetrical. Similar arguments apply to the direct and orthogonal normal models: skew is introduced unless the truncated distribution is symmetrical. Both models could be improved by shifting the angular component so that its modal value lies at $\hat{\theta} = \pi/2$, in the middle of the principal region.² Skew cannot be eliminated completely unless the distribution about the mean happens to be symmetrical, as shown in an idealised form in Figure 7.12, but can be substantially reduced.

When the mean of the orientation features is adjusted in this fashion, it also affects derived fields used by the orthogonal-normal and composite-feature models. Correspondence between the MRF predictor-error distribution and the normal curve is much improved for modified feature data after sampling, as shown for ninth-order Resultant features extracted from the pressed-cork image (#4) in Figure 7.13. Agreement is acceptable, even for images which do not have a predominant orientation (data not shown). A tall central peak remains in the MRF predictor-error distribution from unsampled features, similar to Figure 7.9, but is now sited exactly at the origin (data not shown).

The improvement for the direct normal model is such that one may question whether

²Note that initially $0 \leq \theta < \pi$, but its range is doubled to 2π by the Gabor-Markov models to match the periodicity of trigonometric functions (§6.3.1).

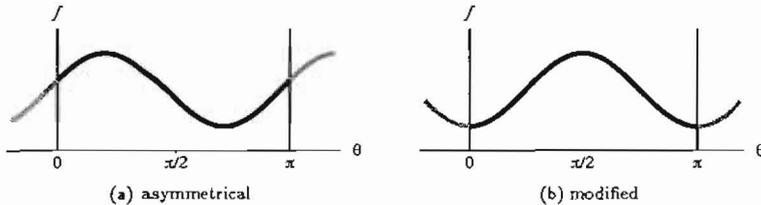


FIGURE 7.12: Illustration of improved Gabor-Markov models. Histograms of Gabor orientation feature θ . (a) typical asymmetric distribution, leading to skew when periodicity of the data is ignored. (b) modified symmetric distribution: mean orientation has been shifted to the centre of the principle segment, and skew is much reduced. We propose transforming the orientation feature in this manner to form improved Gabor-Markov models.

the additional complexity of the orthogonal model is worthwhile. When the angular data is shifted to free the sin component p from skew, it simultaneously introduces maximum skew in the cos field. We could similarly improve agreement with the cosine component q by rotating the mean of the orientation features to $\bar{\theta} = \pi/4$, but this makes one component redundant. Both fields were originally necessary to ensure that the full range of the data was covered with uniform sensitivity but it appears that one component may suffice when the data is shifted to coincide with the most sensitive region of the sinusoid.

Rotating the orientation field has no significant effect on the composite-feature model, with which most of our results were obtained, and which already agrees closely with experimental data. No improvement occurs in the direct fit to the contrast feature since this has no angular component, but some skew is apparent (Figure 7.10b). If pixel values are themselves normally distributed, as appears to hold to a good approximation, we would expect local contrast to follow a normal curve. Our extraction methods (§4.6) fail to obtain the *sign* of the contrast because the contrast feature c is derived from the Gabor energy \mathcal{R} ($\mathcal{R} \sim c^2$), and the positive square root is taken. Skew is introduced when contrast c is modelled by a normal density, but this does not occur with the composite-feature model which is itself derived from c^2 .

7.1.6 Summary of Correspondence with Experimental Data

Poor correspondence was found between dense pixel images and the auto-normal MRF: in all cases, the experimental MRF predictor-error distribution had a taller central peak and longer tails than the normal model, but differences between the fit for the correct and incorrect model were still usually noticeable (§7.1.1). We found that major sources of modelling error are: blur, clipping, and insufficient MRF order. Conventional models are fragile: descriptive accuracy deteriorates quickly if the image is corrupted by noise or blur.

In contrast to the above, the descriptive accuracy of Sampled-Markov models is very high, and remains good after corruption by noise or blur. These differences are reflected in the superior accuracy and robustness of the Sampled-Markov classifiers

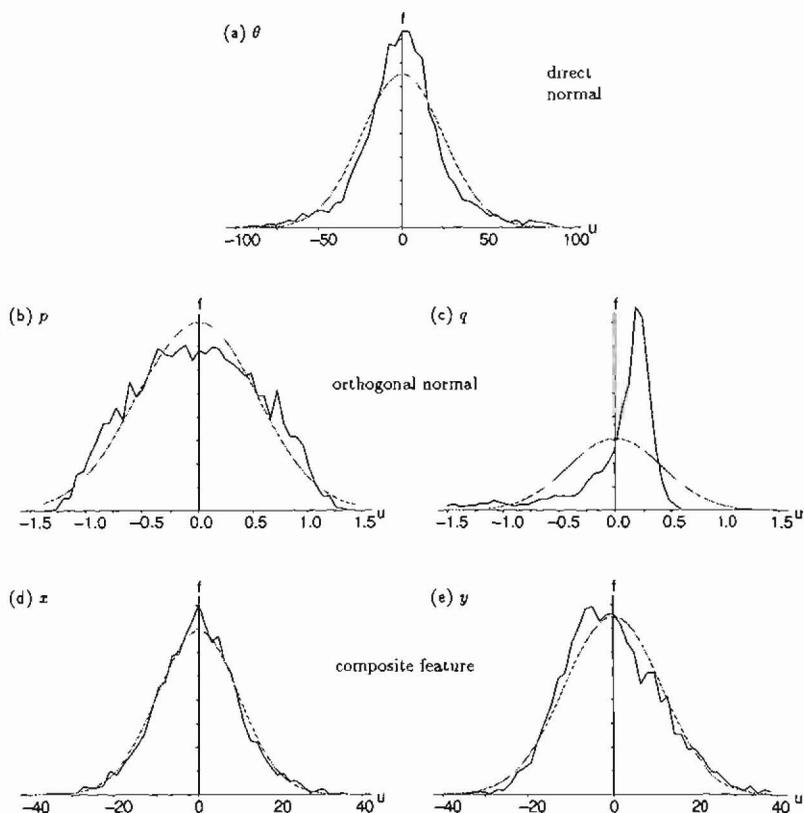


FIGURE 7.13: Agreement between auto-normal and improved Gabor-Markov models. Histograms of observed (solid) and expected (dotted) distributions for ninth-order MRF predictor-error u (arbitrary units), to different vertical scales, using the analysis described in §6.3 and improved in §7.1.5. Gabor features were obtained from the pressed-cork image (#4; Figure 1.6 on page 11) by the Resultant method, and sampled with a period of 4 pixels. Orientation features were rotated to set their mean to $\pi/2$. (a) orientation feature θ and direct normal model; (b,c) orthogonal normal fields model; (d,e) composite-feature model. Agreement in (a,b) is much improved compared to the original orientation features (Figure 7.10), (d,e) are hardly changed, and (c) is much worse because the new mean favours sine rather than cosine.

(§6.7). Sampled models have a higher MRF variance because some information is lost on sampling. Model specificity declines as the sampling period is increased.

Similar trends were observed with MRF predictor-error distributions obtained from unsampled Gabor-Markov models: agreement with the normal curve was poor because the experimental distribution had a tall central peak and long tails, consistent with the effects of image blur. Skew was initially a problem for sampled angular fields but may be substantially reduced by a modified modelling technique (§7.1.5). With this procedure, correspondence between the auto-normal and sampled Gabor-Markov models is much improved, comparable to that achieved by sampled pixel models. Closest agreement was obtained with the composite-feature model, and this was also robust to image corruption. Strong correlation was observed between modelling and classification accuracy, but this link may be broken because very heavy sampling increases the level of agreement with experimental data but reduces specificity.

7.2 Comments on the Influence of Image Blur

Image blur has a strong detrimental effect on classifier accuracy (Chapter 6), and was shown above to decrease the correspondence between the auto-normal MRF model and real data, but its impact is diminished by both the types of sub-sampling we employed above (§6.6). The purpose of this section is to explain these effects, commenting mainly on true-likelihood classification, and to seek a way of reducing misclassification errors introduced by blur.

Blur distorts the image power spectrum, introducing high-contrast peaks which give rise to modelling inaccuracy (§7.2.1). The usual balance between Bayesian penalty and bias terms is upset, prejudicing the classifier towards classes described by models with low MRF variance (§7.2.2). Modified Bayes classifiers have vastly improved robustness with respect to image blur (§7.2.3).

7.2.1 Modelling Error Arising from Image Blur

At first, it seems strange that image blur should introduce modelling error: blur produces strong correlation between adjacent pixels by attenuating high frequencies, but correlation between nearby pixels lies at the heart of Markov random field models (§5.1.2). Following our classification procedure (§6.4.3), we shall model blur by a Gaussian kernel, and will ignore discretisation effects. Let us take an MRF \mathbf{x} and blur it with the kernel $g(b)$:

$$\begin{aligned} \mathbf{y}_i &= \mathbf{x}_i * g_i(b) \\ Y_i &= X_i G_i(b) \end{aligned} \quad (7.1)$$

$$\begin{aligned} g_{m+nM}(b) &= \frac{1}{2\pi b^2} \exp\left[-\frac{1}{2b^2}(m^2 + n^2)\right] \\ G_{m+nM}(b) &= \exp\{-2\pi^2 b^2 [\Xi^2(m/M) + \Xi^2(n/N)]\} \end{aligned} \quad (7.2)$$

where

$$\Xi(f) = \begin{cases} f & \text{if } f \leq 1/2 \\ 1-f & \text{otherwise} \end{cases} \quad (7.3)$$

and the subscripts $m+nM$ reflect raster-ordering of the image array. We introduce the "spectral mapping function" $\Xi(\cdot)$ to convert from lattice to frequency co-ordinates at which the transformed kernel $G \Rightarrow g$ is evaluated. Our aim is to express the new field \mathbf{y} as an auto-normal MRF. If this may be done, the expected power spectrum $\mathcal{E}\{|\mathbf{Y}|^2\}$ will assume the usual form over the image lattice \mathcal{L} :

$$\begin{aligned} \mathcal{E}\{|\mathbf{Y}_i|^2\} &= \mathcal{E}\{|\mathbf{X}_i|^2 G_i^2(b)\} \\ &= \frac{(\sigma^2)_x}{1-B_{x,i}} G_i^2(b) \\ &\stackrel{?}{=} \frac{(\sigma^2)_y}{1-B_{y,i}}, \quad \forall i \in \mathcal{L} \end{aligned} \quad (7.4)$$

where we take $(\sigma^2)_x$ to refer to the *model* variance for parameter set \mathcal{G}_x , and B_x refers to the cosine transform of the MRF parameter vector β_x (5.63), and similarly for $(\sigma^2)_y$ and B_y . This equation may be solved trivially for $(\sigma^2)_x = 0$ (constant field) but does not otherwise give \mathcal{G}_y of finite order. Perhaps this is not surprising with reference to the Markov property (§5.1.2), because blur has distributed the necessary pixel values over a larger area so that the required neighbourhood set is no longer given. Only if the entire field is known can this information be recovered. Gaussian blur is not spatially limited but a similar result is obtained for a finite kernel. Our conclusion is that after applying blur, a non-trivial auto-normal MRF is no longer an auto-normal MRF of finite order.³

Whereas the blurred field may not be represented *exactly* as an auto-normal MRF, this is not of crucial importance in practical cases because Markovianity of the original field is itself only approximate, and it is certainly possible to estimate MRF parameters from a blurred image. Power spectra of blurred images tend to be heavily concentrated in the low frequencies around the origin because higher frequencies are attenuated. The model spectrum is formed from the reciprocal of periodic components:

$$\begin{aligned} \mathcal{E}\{|\mathbf{X}_{m+nM}|^2\} &= \sigma^2 / \lambda_{m+nM} \\ &= \frac{\sigma^2}{1 - 2 \sum_{(k,l) \in \mathcal{N}^+} \beta_{k+lS} \cos[2\pi(mk/M + nl/N)]} \end{aligned} \quad (7.5)$$

where λ_i are spectral density coefficients, and S represents the span of the MRF neighbourhood \mathcal{N}^+ . In order to obtain the peak near the origin, it is necessary for λ_i to approach zero, but often λ_i becomes so close to zero that any small change in any element of the MRF parameter vector β is sufficient to break the homogeneity constraint (5.66 on page 115; Figure 6.1 on page 140), giving $\lambda_i \leq 0$ for some frequency. In practice, the freedom for variation of the coefficients is so small that full 15-figure decimal accuracy must be preserved. The model is very unstable, since $1/\lambda_i$ is clearly sensitive

³We have not attempted to obtain a stronger version of this result as our primary concern is with the auto-normal MRF.

to numerical error when $\lambda_i \geq 0$, and there is reduced freedom of choice to fit the data closely when the neighbourhood coefficients β are meshed together so closely.

A further practical difficulty is felt by the parameter estimation algorithm. The pseudo-likelihood parameter vector $\hat{\beta}$ is used to form an initial estimate of the true-likelihood parameter vector $\hat{\beta}_0$ used to seed our gradient-ascent parameter-estimation algorithm (§6.2.1). The seed point $\hat{\beta}_0$ almost always has to be revised, however, because the true likelihood maximum lies very close to a forbidden (inhomogeneous) region, into which $\hat{\beta}$ usually falls. Once a valid starting point has been found, gradient ascent may only proceed extremely slowly because parameters can only change by a tiny amount at each iteration without leaving the permitted region. Estimating the MRF parameters from a blurred image may take 50 or more times as long as for the original image.⁴

Blur is not unique in causing modelling difficulty: any transformation causing the power spectrum to contain large peaks will similarly affect the parameter estimates. It is particularly easy to introduce blur, however, and we have shown that even a small amount can have drastic practical consequences (§6.5).

7.2.2 Classification Error Arising from Image Blur

Whatever the effect of blur on modelling, it certainly has a strongly adverse effect on classification accuracy (§6.5.3): even a moderate amount of blur, space-constant $b = 0.5$ pixels, is sufficient to increase the error rate of the fifth-order dense pixel classifier from 6.8% to 84%. In order to explain this, we shall consider the effect of image blur on the true log-likelihood function, L_b :

$$2L_b(\mathbf{x} * g(b)) = \sum_{i \in \mathcal{L}} \log \lambda_i - MN \log(2\pi\sigma^2) - \sum_{i \in \mathcal{L}} \frac{|X_i|^2 G_i^2(b)\lambda_i}{\sigma^2} \quad (7-6)$$

where summation for i extends over the dimensions of the image lattice \mathcal{L} (compare 6.13 on page 146). The Gaussian kernel $G(b)$ was given in (7.2), and satisfies: $G(b) \leq 1$. The first two terms of (7-6) are unaffected by blur, and the third term, which contains all the image information, is always made less negative. Hence, the effect of blur is always to increase the likelihood L_b . In the limit as the blur space-constant increases to infinity ($b \rightarrow \infty$), the blurred image assumes the same value everywhere, and the likelihood reaches its maximum equal to the sum of the first two terms of (7-6), and is independent of the original texture:

$$\begin{aligned} 2L_{\max} &= \sum_{i \in \mathcal{L}} \log \lambda_i - MN \log(2\pi\sigma^2) \\ 2L_{b,X} &= \sum_{i \in \mathcal{L}} \frac{|X_i|^2 G_i^2(b)\lambda_i}{\sigma^2} \end{aligned} \quad (7-7)$$

where $L_b = L_{\max} - L_{b,X}$. The effect of blur is to introduce bias towards the class for which L_{\max} is the largest, and this term is often dominated by the MRF variance, σ^2 .

⁴Estimation of ninth-order true-likelihood parameters from the pressed-cork image (#4) required 302 iterations of the gradient-ascent algorithm. When the image was first blurred by convolution with a Gaussian, space-constant 1 pixel, this increased to 9641 iterations, a factor of 32.

Usually, a large MRF variance results in a small bias term L_{\max} , which is balanced by a smaller image error penalty term $L_{b,X}$ (since this is divided by σ^2). Similarly, a model with low variance usually has a larger bias term but also a correspondingly greater image penalty term. Blur always reduces the image penalty term $L_{b,X}$, predisposing the classifier towards classes with a large bias term L_{\max} and hence a small MRF variance. As the amount of blur is increased, so the influence of the original data diminishes.

In terms of empirical classification accuracy, we predict that blurred images whose true texture model has a high MRF variance will be misclassified, and that models with low MRF variance will generate false-positives. On comparing confusion matrices for uncorrupted and blurred images (Figures 7.14 and 7.15), both of which have been sorted by MRF variance (Figure 7.16), we find that *all* misclassification errors involve a shift to lower variance, appearing below the leading diagonal, consistent with the above prediction. The degree of blur (parameter $b = 0.5$ pixels) was small enough for the image data to retain some influence, and most errors were from large to medium variance, but bias towards models of smaller variance becomes stronger as the blur parameter is increased (data not shown). Note that because the Sampled-Markov classifiers achieve acceptable error rates using blurred images (§6.6), there is not a problem of insufficient information (see also Figure 6.5 on page 154).

The rate at which blur distorts an image in practice clearly varies with the image itself. We shall consider the expected effect of blur on pixel and error variance terms for a true MRF evaluated with the correct parameter set (§A.5.1):

$$\begin{aligned} V_b(u) &= \frac{\sigma^2}{MN} \sum_{i \in \mathcal{L}} (1 - B_i) G_i^2(b) \\ V_b(x) &= \frac{\sigma^2}{MN} \sum_{i \in \mathcal{L}} \frac{G_i^2(b)}{(1 - B_i)} \\ 2L_{b,X} &= \sum_{i \in \mathcal{L}} G_i^2(b) \end{aligned} \quad (7-8)$$

where $V_b(u)$ and $V_b(x)$ represent variances of the MRF predictor-error and pixel data respectively, $L_{b,X}$ is the likelihood penalty term introduced by (7-7), and summation for i extends over the dimensions of the image lattice \mathcal{L} ($M \times N$ pixels). Typically for our Brodatz images, the spectral density coefficient $\lambda_i = 1 - B_i$ is very small near the origin (≈ 0.001), suggesting that MRF predictor-error variance $V_b(u)$ will decrease very rapidly for blurred images because the high frequencies which contribute most to this term are heavily attenuated. Pixel variance $V_b(x)$ is less affected because it receives the greater contribution from low spatial frequencies which are relatively unaffected by blur. The penalty term $L_{b,X}$ used by the true-likelihood classifier (7-7) is intermediate between these two cases when the field is consistent with the MRF parameter set, and when the blur parameter b is large enough for aliasing to be insignificant (§A.5.1), $L_{b,X} \propto (4\pi b^2)^{-1}$. The effect of blur is more complicated when the model does not accurately describe the field. In particular, an image dominated by high spatial frequencies or a model dominated by low spatial frequencies will be affected more rapidly than (7-8).

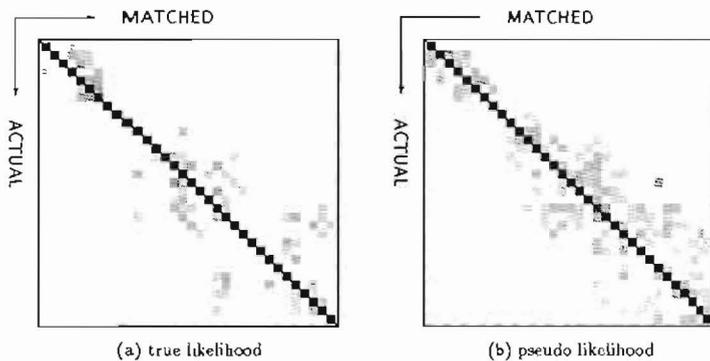


FIGURE 7.14: Scatter matrices for uncorrupted pixel classifiers. Our uncorrupted Brodatz image set was classified by the All-Quarters method, described in §6.4.3. The matrices have been ordered by increasing MRF variance (33 classes) and are pooled from models of order 1–5, 7 and 9. Correctly classified images lie along the leading diagonal, and intensity is proportional to frequency. (a) true-likelihood; (b) pseudo-likelihood. No obvious bias towards higher or lower variance is present.

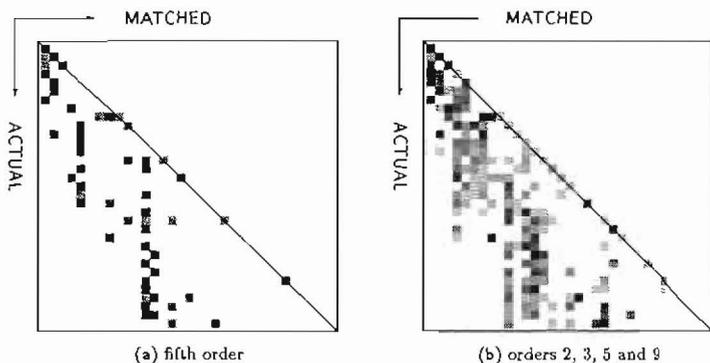


FIGURE 7.15: Scatter matrices for pixel classifiers with blur. Our Brodatz image set was blurred with parameter 0.5 pixels and classified using true-likelihood parameter sets by the All-Quarters method, described in §6.4.3. The matrices have been ordered by increasing MRF variance. Correctly classified images lie along the leading diagonal, and intensity is proportional to frequency. (a) fifth-order classifier; (b) classifiers of order 2, 3, 5 and 9. In both cases, every misclassification is caused by the selection of a model of lower variance, appearing here as a shift below the diagonal.

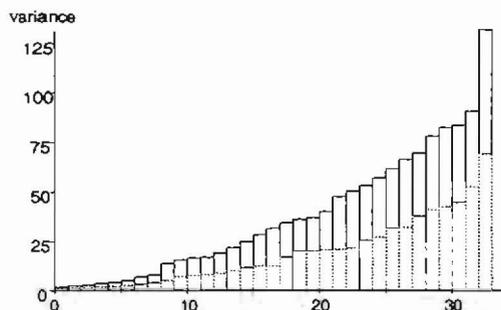


FIGURE 7.16: Histograms of MRF variance for unsampled pixel models. Parameter sets were estimated from our Brodatz image set using true-likelihood estimation (§6.2.1). Sorted cumulative histograms of MRF variance (gray-level units squared) are shown. *Solid*: fifth order; *Dotted*: ninth order. Average model variance is a decreasing function of MRF order (Figure 7.3).

7.2.3 Measures to Counteract Image Blur

Image Restoration

The classical approach to image restoration is Wiener filtering (§5.5.2). The given image y , assumed degraded by blur, is effectively de-convolved by dividing out the transform spectrum:

$$\hat{X}_i = Y_i/G_i(b) \quad (7.9)$$

where \hat{X} is the Fourier transform of the restored image (Rosenfeld & Kak, 1982). This method can be effective but steps must usually be taken to ensure that noise does not dominate for frequencies where the blur kernel $G(b)$ is small, and the blur parameter b must be known in advance.

Modified Bayes Classifier

Our main objective is to obtain a measure of immunity from classification error by increasing robustness. It is tempting to seek to modify the data to fit the model by dividing out the blur spectrum, forming a modified penalty term L'_b :

$$2L'_b(\mathbf{x}; b) \stackrel{?}{=} \sum_{i \in \mathcal{L}} \log \lambda_i - MN \log(2\pi\sigma^2) - \sum_{i \in \mathcal{L}} \frac{|X_i|^2 \lambda_i}{\sigma^2 G_i^2(b)} \quad (7.10)$$

where the original form for L_b is given by (7.7). We might hope to maximise $L'_b(\mathbf{x}; b)$ with respect to b to undo the transformation and simultaneously estimate the blur parameter. Unfortunately, just as blurring the image always increases the likelihood, de-blurring it will always decrease it, causing this approach to fail.

A more profitable approach is to use a modified Bayes classifier (§5.4.1), where the model is adjusted to fit the data, forming the new likelihood function P'_b :

$$P'_b(\mathbf{x}; b) = \prod_{i \in \mathcal{L}} \left(\frac{1}{2\pi S_{x_i} G_i^2(b)} \right)^{\frac{1}{2}} \exp \left[-\frac{1}{2} \sum_{i \in \mathcal{L}} \frac{|X_i|^2}{S_{x_i} G_i^2(b)} \right] \quad (7.11)$$

where $S_{x_i} = \sigma^2 \lambda_i^{-1}$, and summation for i extends over the dimensions of the image lattice \mathcal{L} (compare 5.64 on page 114). The modified likelihood $P'_b(\mathbf{x}; b)$ may now safely be maximised with respect to b to overcome the effect of the blur transform. Classification is based on these maximum likelihoods, Π_j :

$$\begin{aligned} \Pi_j &= \max_b \mathcal{P}\{\omega_j | \mathbf{x}, b\} \\ \hat{\omega}(\mathbf{x}) &= \omega_i, \text{ if } \Pi_i = \max_j \Pi_j \end{aligned} \quad (7.12)$$

and an estimate \hat{b} is also obtained for the degree of blur. A similar approach has recently been applied to other viewing parameters (Cohen *et al.*, 1991), and a simpler technique was suggested to accommodate global changes in illumination (Kashyap *et al.*, 1982).

As was mentioned above (§7.2.1), the blurred MRF no longer has finite order, but this is of no practical consequence since the use of spectra avoids the need for direct convolution. The modified likelihood (7.11) is easily decomposed:

$$\begin{aligned} 2L'_b(\mathbf{x}; b) &= \sum_{i \in \mathcal{L}} \log [\lambda_i G_i^{-2}(b)] - MN \log(2\pi\sigma^2) - \sum_{i \in \mathcal{L}} \frac{|X_i|^2 \lambda_i}{\sigma^2 G_i^2(b)} \\ \frac{\partial L'_b}{\partial b} &= 4\pi^2 b \left(1 - \sum_{i \in \mathcal{L}} \frac{|X_i|^2 \lambda_i}{\sigma^2 G_i^2(b)} \right) \sum_{(m,n) \in \mathcal{L}} \Xi^2(m/M) + \Xi^2(n/N) \end{aligned} \quad (7.13)$$

where L'_b is the log-likelihood derived from (7.11), and $\Xi(\cdot)$ is the spectral mapping function (7.3). Unfortunately, applying the usual criterion $\partial L'_b / \partial b = 0$ does not usually lead to an analytic form for b ; but because $\lambda_i > 0$, the gradient is a decreasing function of b , and a numerical algorithm converges rapidly. If $\partial L'_b / \partial b|_{b=0} \geq 0$, it is immediately clear that $\hat{b} = 0$.

This approach is somewhat idealised because we have taken no account of aliasing, quantisation error or other noise effects. It does not depend on the blur being Gaussian, although this is convenient, since any spectrum may be substituted for $G(b)$, and a different model would be preferred if more realistic transformation parameters were known. Noise becomes a problem when the blur space-constant b is large, and in these circumstances it might be sensible to consider only the low-frequency part of the spectrum, to cap $G_i^{-2}(b)$, or to adopt other corrective measures (Rosenfeld & Kak, 1982).

Preliminary Results with Modified Classifier

Early results with the modified classifier described above have been encouraging. When we used it to estimate the (known) amount of blur present in two synthetic images, blurred according to §6.4.3, the blur parameter was estimated accurately (Table 7.1). Interestingly, the classifier also recorded the presence of blur in our original Brodatz

blur parameter b	estimated blur parameter \hat{b}	
	#4 pressed cork	denim
0.0	0.00	0.00
0.2	0.00	0.07
0.4	0.35	0.38
0.6	0.59	0.61
0.8	0.81	0.82
1.0	1.01	1.02
1.2	1.22	1.22
1.4	1.27	1.26

TABLE 7.1: Image blur estimated by the modified classifier. Ninth-order 256×256 synthetic textures for the pressed-cork and denim images (originals in Figures 6.2 and 4.13) were blurred with parameter b (pixel units). Except for b large or small, when noise and aliasing are most significant, the estimated blur parameter obtained by the modified classifier is very close to the true value.

images (data not shown), consistent with their poor correspondence with the auto-normal model (Figure 7.1). A possible extension would be to allow blur as a free parameter during the original parameter-estimation stage as this would permit the underlying texture variation to be captured more closely. If the modified classifier is used to classify images degraded other than by blur, errors may result.

Using the modified classifier, we repeated the classification exercise for Brodatz images corrupted by moderate synthetic blur (parameter $b = 0.5$ pixels) using true-likelihood fifth-order parameter sets. The error rate fell dramatically from 84% with the original classifier (§6.5.3) to 9.8%, with estimates for the blur parameter bunched tightly around 0.5 ($\hat{b} = 0.505 \pm 0.06$ pixels). Although the synthetic blur applied here was close to the form assumed by the model, this is a significant improvement in classification accuracy with real images which confirms the validity of the approach.

7.3 Comments on the Influence of Additive Noise

Our classification results show that the addition of a small amount of noise compared to the inherent variation of each pixel has a strongly adverse effect on unsampled pixel models (§6.5.2), affecting Gabor-Markov and sampled pixel models to a lesser degree (§6.6). It is surprising at first that additive random noise should cause problems for an MRF model since it is built around the random variation of each pixel. Our goal is to examine the interaction of additive noise with the auto-normal model, and hence to suggest ways of restricting its unwelcome consequences. As above, we shall distinguish between modelling and classification error, and show that noise prejudices the classifier towards classes described by models with high MRF variance (§7.3.2). We propose a modified classifier which may arrest this trend (§7.3.3).

7.3.1 Modelling Error Arising from Additive Noise

Modelling error refers to the extent of disagreement between a parameter set and the image from which it was estimated. In §7.1.1, we found experimentally that an image to which white noise had been added, following our simple noise model (§6.4.3), was no longer described accurately by an auto-normal MRF. In order to isolate modelling error attributable to additive noise, let us consider taking an MRF and adding independent white noise of variance ρ^2 , forming the field $\mathbf{x} = \mathbf{X}$:

$$X_i = \nu \sqrt{\frac{\sigma^2}{1 - B_i}} + \rho \nu' \quad (7.14)$$

where ν and ν' are independent zero-mean unit-variance normal random variables, and \mathbf{B} is the cosine transform of the MRF parameter vector β (5.63). We may combine the terms by adding variances:

$$X_i = \nu \sqrt{\left(\frac{\sigma^2}{1 - B_i} + \rho^2\right)}. \quad (7.15)$$

We may now attempt to find an MRF \mathcal{G}_x which gives the correct form for the expected power spectrum of the new field:

$$\begin{aligned} \mathcal{E}\{|X_i|^2\} &= \frac{\sigma^2}{1 - B_i} + \rho^2 \\ &\stackrel{?}{=} \frac{(\sigma^2)_x}{1 - B_{x,i}}, \quad \forall i \in \mathcal{L} \end{aligned} \quad (7.16)$$

where we again use $(\sigma^2)_x$ to denote the *model* variance for parameter set \mathcal{G}_x . Unfortunately, just as when the image is corrupted by blur (§7.2.1), this has no general solution except in trivial cases. This is again consistent with the idea of the Markov neighbourhood: now that these values are no longer known exactly, additional information may be obtained from the wider surround. It is possible to ignore this and obtain the best estimate possible with the given number of parameters. The amounts of noise used in our experiments were much less than the pixel variance, and hence noise was not a source of modelling error of practical concern.

7.3.2 Classification Error Arising from Additive Noise

Noise has a strong adverse impact on classification accuracy (§6.5.2), particularly for unsampled models. We shall examine the expected effect on the likelihood function of corruption by additive noise, evaluating the true likelihood with the original parameter set \mathcal{G}_x but the noisy field $\mathbf{x} + \rho\nu$:

$$\begin{aligned} 2L_n(\mathbf{x} + \rho\nu) &= \sum_{i \in \mathcal{L}} \log \lambda_i - NM \log(2\pi\sigma^2) - \sum_{i \in \mathcal{L}} \frac{(|X_i|^2 + \rho^2)\lambda_i}{\sigma^2} \\ &= 2L(\mathbf{x}) - \rho^2 \sum_{i \in \mathcal{L}} \lambda_i / \sigma^2 \\ &= 2L(\mathbf{x}) - MN\rho^2 / \sigma^2 \end{aligned} \quad (7.17)$$

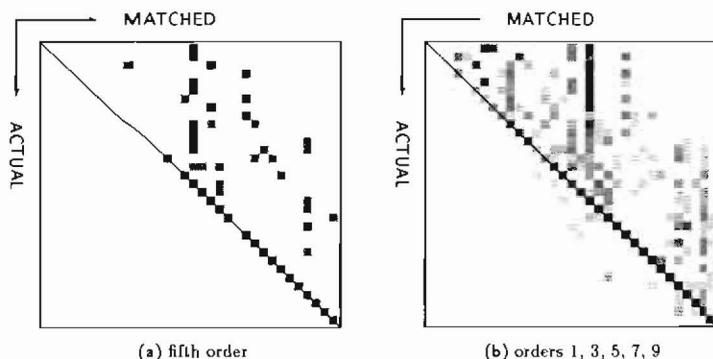


FIGURE 7.17: Scatter matrices for pixel classifiers with noise. Our Brodatz image set was classified using true-likelihood parameter sets by the All-Quarters method (§6.4.3) after addition of noise of variance 20 gray-level units squared. The matrices have been ordered by increasing MRF variance. Correctly classified images lie along the leading diagonal, and intensity is proportional to frequency. (a) fifth-order classifier; (b) classifiers of order 1, 3, 5, 7 and 9. Almost all misclassification errors are caused by the selection of a model of *higher* variance, appearing here as a shift above the diagonal.

where $L(\mathbf{x})$ is the original log-likelihood (no noise), given by (6-13) on page 146. An extra term appears, and since $\lambda_i > 0$, its effect is always to *decrease* the likelihood. As above (§7.2.2), we shall separate bias and penalty terms, writing: $L_n = L_{\max} - L_{n,x}$. We may compare the expected impact of noise on MRF predictor-error and pixel variances (§A.5.1):

$$\begin{aligned} V_n(u) &= \sigma_u^2 + \rho^2(1 + 2\|\beta\|^2) \\ V_n(x) &= \sigma_x^2 + \rho^2 \\ 2L_{n,x} &= MN(1 + \rho^2/\sigma^2) \end{aligned} \quad (7-18)$$

where σ_u^2 and σ_x^2 refer to the original predictor-error and pixel variances, respectively. The pixel variance $V_n(x)$ increases linearly with noise variance, but the predictor-error variance $V_n(u)$ increases more rapidly, and proportionally by a much higher rate (since $\sigma_u^2 < \sigma_x^2$). Further, the increase will probably be larger for models of higher order because of the weighting from $\|\beta\|^2$. The classifier penalty term $L_{n,x}$ increases at a proportionately greater rate for small MRF variance σ^2 , upsetting the balance between bias and penalty terms in a similar manner to image blur (§7.2.2), and prejudicing the classifier. In this case, the trend is towards models of higher MRF variance, confirmed by the scatter matrix shown in Figure 7.17, where almost all misclassifications involve the selection of a model of higher variance. This appears as a shift above the leading diagonal (compare uncorrupted images in Figure 7.14), and is the opposite effect to blur (Figure 7.15). A clear trend in the results for the dense classifier was for the error rate under noise to increase for models of larger order (Figure 6.16 on page 168), and we may now explain this counter-intuitive effect. When the model order is large, the MRF predictor-error variance σ^2 is small (Figure 7.3), increasing the impact of external noise according to the factor ρ^2/σ^2 in (7-18).

7.3.3 Measures to Counteract Noise

Image Restoration

Noise is not deterministic, and so restoration cannot hope to restore the original image exactly, but instead seeks to remove the *expected* effect of the noise. In the case of the Wiener filter (§5.5.2), noise adds a constant amount to the power spectrum and so this is simply subtracted to give the restored image:

$$\hat{X}_i = \frac{S_{y,i} - \rho^2}{S_{y,i}} Y_i \quad (7.19)$$

where \mathbf{Y} is the Fourier transform of the degraded image \mathbf{y} , and $\hat{\mathbf{X}}$ is the transform of the restored image (Rosenfeld & Kak, 1982). This method requires the noise variance ρ^2 to be known in advance. In our case, the form of the restored image may be assumed, $S_{x,i} = \sigma^2/\lambda_i$, and this may be substituted into (7.19):

$$\hat{X}_i = \frac{1}{1 + \lambda_i \rho^2 / \sigma^2} Y_i \quad (7.20)$$

Maximum *a posteriori* (MAP) restoration is more flexible and is generally preferable. We maximise the likelihood of the original image \mathbf{x} given the observed degraded copy \mathbf{y} :

$$\begin{aligned} P(\mathbf{x} | \mathbf{y}) &\propto P(\mathbf{y} | \mathbf{x})P(\mathbf{x}) \\ P(\mathbf{y} | \mathbf{x}) &= \prod_{i \in \mathcal{C}} G(x_i - y_i, \rho^2). \end{aligned} \quad (7.21)$$

In this case, the form of $P(\mathbf{x})$ is assumed known. The likelihood must usually be maximised numerically to find $\hat{\mathbf{x}}$.

Modified Bayes Classifier

Following the approach taken in §7.2.3, we may design a modified classifier that contains the amount of added noise as a free parameter. It is again tempting to modify the data to fit the model, forming the new likelihood P'_n (from 5.64 on page 114):

$$P'_n(\mathbf{x}; \rho) \stackrel{?}{=} \prod_{i \in \mathcal{C}} \left(\frac{1}{2\pi S_{x,i}} \right)^{\frac{1}{2}} \exp \left[-\frac{1}{2} \sum_{i \in \mathcal{C}} \frac{|X_i|^2 - \rho^2}{S_{x,i}} \right] \quad (7.22)$$

but subtracting noise from an image in this way always increases the likelihood and so it is impossible to maximise $P'_n(\mathbf{x}; \rho)$ to obtain an estimate $\hat{\rho}$. The solution is again to modify the model to fit the data:

$$P'_n(\mathbf{x}; \rho) = \prod_{i \in \mathcal{C}} \left(\frac{1}{2\pi(S_{x,i} + \rho^2)} \right)^{\frac{1}{2}} \exp \left[-\frac{1}{2} \sum_{i \in \mathcal{C}} \frac{|X_i|^2}{S_{x,i} + \rho^2} \right] \quad (7.23)$$

forming a modified Bayes classifier (§5.4.1). It is again of no practical consequence that this equation no longer represents an MRF of finite order. The modified log-likelihood L'_n

arising from (7.23) is easily decomposed:

$$\begin{aligned}
 2L'_n(\mathbf{x}; \rho) &= \sum_{i \in \mathcal{C}} \log \lambda_i - \sum_{i \in \mathcal{C}} \log(\sigma^2 + \rho^2 \lambda_i) - \sum_{i \in \mathcal{C}} \frac{|X_i|^2 \lambda_i}{\sigma^2 + \rho^2 \lambda_i} \\
 \frac{\partial L'_n}{\partial \rho} &= \rho \sum_{i \in \mathcal{C}} \frac{\lambda_i (\lambda_i |X_i|^2 - \sigma^2 - \rho^2 \lambda_i)}{(\rho^2 \lambda_i + \sigma^2)^2} \\
 \left. \frac{\partial L'_n}{\partial \rho} \right|_{\rho=\hat{\rho}} &= 0
 \end{aligned} \tag{7.24}$$

but the noise estimate $\hat{\rho}$ can only be found numerically, apart from the trivial solution $\hat{\rho} = 0$. This procedure has not currently been implemented.

7.4 Comments on the Influence of Image Sampling

Sub-sampling of both types (selection and averaging) improves classification accuracy for corrupted and uncorrupted data (§6.6). This phenomenon is of practical importance, and several aspects of sampling are investigated in this section. Only the power spectrum of the test image is used during true-likelihood classification (§6.4), and hence any explanation demands consideration of the effect of sub-sampling on the image spectrum. For clarity, some of the results below are given for a 1-D signal but they extend easily into two dimensions. We contend that the increased modelling accuracy of Sampled-Markov models arises from changes in the morphology of the image spectrum (§7.4.2). A reduction in the contrast of spectral peaks brings the data into closer correspondence with our assumed statistical model. Sampling reduces the effective magnitude of both image noise and blur, enhancing classifier robustness (§7.4.3; §7.4.4).

7.4.1 Spatial-Frequency Spectra and Aliasing

Unsampled images are necessarily band-limited, $-\pi < \omega \leq \pi$, since they are already expressed as a discrete array, but aliasing still occurs on sampling (§A.5.2). This is visualised for a sampling period $s = 1.5$ pixels in Figure 7.18: the contents of the heavy box form the spectrum of the sampled image, which is contaminated by aliasing in the shaded regions where adjacent copies of the spectrum overlap. Aliasing causes unpredictable distortion of the signal according to the phase, and can be a significant source of error. Smooth-sampling reduces aliasing error by partially attenuating high frequencies: adjacent copies of the frequency spectrum still overlap to the same degree, but the aliased regions contain a lower proportion of the image energy. Our simple window function is far from being an ideal band-pass filter (Meer *et al.*, 1987) because its transfer function is insufficiently abrupt at the cut-off frequency (Figure A.9*b* on page 273). The effect of anti-aliasing filters is discussed further below (§8.3).

According to the diagram shown in Figure 7.18, sampling magnifies the proportion of the image spectrum occupied by the energy formerly concentrated in the low-frequency band. Coupled with the effects of aliasing, this tends to even out the distribution of energy across spatial frequencies in the sampled image. Power spectra of the dense and

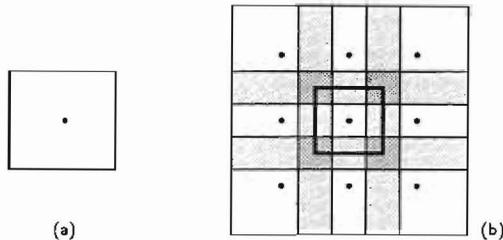


FIGURE 7.18: Illustration of frequency aliasing arising from sampling. Each box represents the range of spatial frequencies occupied by a discrete two-dimensional image with the origin, shown here by a dot, at its centre. (a) unsampled image: no aliasing. Copies of the spectrum effectively abut but do not overlap. (b) sampled image, period 1.5 pixels: adjacent copies of the spectrum overlap (shaded regions), causing frequency aliasing. The smaller heavy box represents the spectrum of the sampled image. Energy at high frequencies in the original image is mapped to new frequencies in the sampled image.

sampled pressed-cork image (#4; Figure 1.6 on page 11) are shown in Figure 7.19: a dominant feature of the unsampled spectrum is a large peak near the origin, shown here at any corner of the array, reflecting the strong contribution from low spatial frequencies. Peak contrast is lower in sampled images reflecting both truncation of the frequency range and aliasing effects (*b-d*). As shown by comparison of (*b*) and (*c*), this change is more gradual after smooth-sampling because the window kernel transmits the low-frequency band which usually contains the peak, but attenuates higher frequencies. When the sampling period is too large, the spectrum becomes unrecognisable (*d*), due to truncation of the frequency range and severe aliasing.

7.4.2 Effects of Sampling Uncorrupted Images

In §7.2, we argued that the presence of high-contrast peaks in the power spectrum created both theoretical and practical difficulty for the MRF model, and the ubiquity of these features in our Brodatz set is illustrated by the pressed-cork image (Figure 7.19). Sub-sampling reduces the contrast of the spectral peak, and hence should improve modelling accuracy: exactly the trend noted in §7.1.3, and consistent with a lower baseline error for sub-sampled images (§6.6). A conflict exists: too much sampling, and the spectrum will be easy to model accurately but will contain insufficient information to discriminate texture classes reliably; too little, and the necessary information will be present but a badly-fitting model will be unable to make full use of it. Our current sampling techniques cause aliasing error, which has an unpredictable effect on the shape of the power spectrum but increases in magnitude for larger sampling period.

The tendency of sub-sampling to reduce the spectrum peak contrast is demonstrated more quantitatively by Figure 7.20: we examined the minimum value of the spectral density coefficient λ_{\min} using a fifth-order classifier, averaged across all our 132 Brodatz images. As predicted, λ_{\min} is very small for the unsampled image set (sampling period $s = 1$ pixel), corresponding to large peaks in the power spectrum, which are proportional

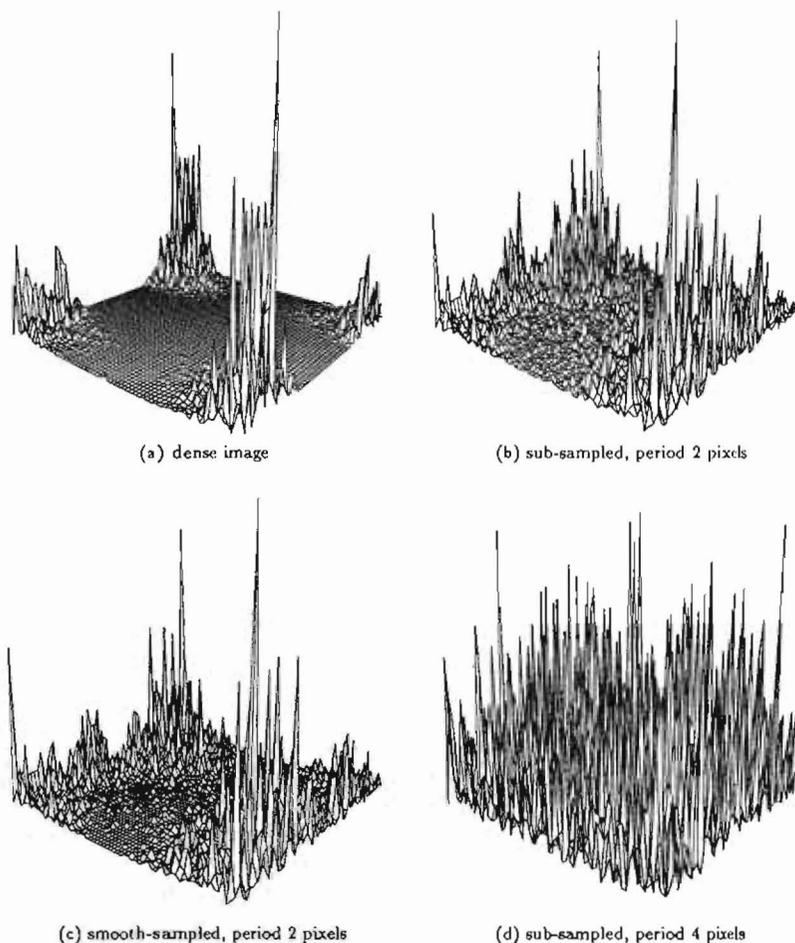


FIGURE 7.19: Power spectra of dense and sampled Brodatz images. Horizontal axes represent spatial frequency, with the origin at any corner, and height represents image energy. (a) power spectrum of the pressed-cork image (#4; Figure 1.6): note the concentration of energy at low spatial frequencies. (b,c) after sampling and smooth-sampling, respectively, with a period of two pixels. In both cases, the spectral peaks are relatively broader and have reduced contrast. (d) after sampling with a period of four pixels.

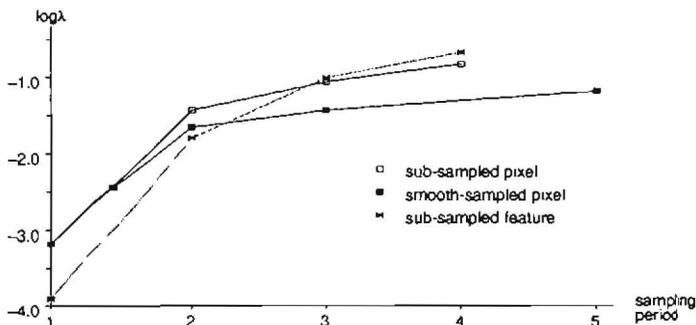


FIGURE 7.20: Effect of sampling on contrast of spectral peak. Minimum spectral density coefficient λ_{\min} , shown to a \log_{10} scale, averaged across our 132 Brodatz image set for a fifth-order true-likelihood classifier. *Solid*: pixel classifier, smooth- and sub-sampled; *Dotted*: sub-sampled Resultant composite-feature classifier. When λ_{\min} is small, high peak contrast results, causing difficulty for parameter estimation algorithms.

to $1/\lambda_{\min}$. As the sampling period is increased, peak contrast progressively diminishes although the central peak remains (shown here to a \log_{10} scale). Smooth-sampled spectra tend to have a somewhat taller peak than for selection sampling, consistent with the effect of the window function in preserving peak contrast, but this appears to be more than offset in terms of classification accuracy by a reduction in aliasing error.

It is difficult to predict the general effect of sub-sampling on classification accuracy because it is closely influenced by the actual spectra. If the dense image fits the MRF model closely, sub-sampling is liable to reduce the amount of information and add aliasing error without any clear theoretical benefit, but if our experience is typical, sub-sampling has much to offer in practice, both in terms of fidelity of representation and computational efficiency. Reducing the dimensions of the image by a factor n decreases the computational requirement by a factor greater than n^2 during classification. Parameter estimation benefits to a greater degree because the relative absence of meshing of coefficients means that far fewer iterations are required for the gradient-ascent algorithm (§7.2.1).⁵

The expected power spectrum corresponding to a particular parameter set may be computed by taking the cosine transform of MRF neighbourhood coefficients (7.5 on page 195). Note that this procedure is distinct from measuring the power spectrum of a texture synthesised from these parameters because random fluctuations are eliminated. Compare real power spectra (Figure 7.19a,b) with those derived from ninth-order true-likelihood parameter sets (Figure 7.21), both obtained from the pressed-cork image (#4; Figure 1.6), and shown from the same "viewpoint" but to different vertical scales. Superficially, the model spectrum of the unsampled image (Figure 7.21a) has the correct

⁵ Estimation of ninth-order true-likelihood parameters from the unsampled pressed-cork image (#4) required 302 iterations, falling to 169 after sampling with a period of 2 pixels, and further to 104 iterations for a period of 4 pixels.

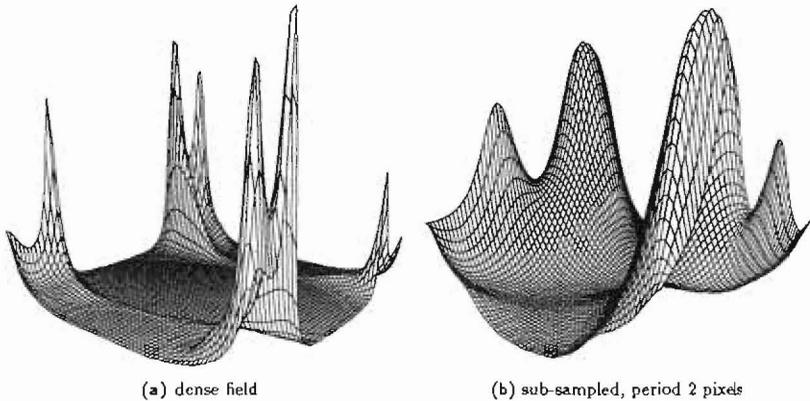


FIGURE 7.21: Model power spectra of dense and sampled Brodatz images. Expected power spectra corresponding to ninth-order true-likelihood MRF parameter sets estimated from the pressed-cork image (#4; Figure 1.6). Horizontal axes represent spatial frequency, with the origin at any corner, and height represents image energy. (a) unsampled model; (b) sub-sampled, period 2 pixels. Compare the real spectra (Figure 7.19a,b).

form, containing peaks at low frequencies and a central plateau. We believe that assuming this general form imposes strain on the model, so that there is little remaining freedom to follow the contours of the real spectrum closely. After sampling (period 2 pixels), the form of the spectrum is more undulating (Figure 7.21b), and is more compatible with the cosine variation from which it is derived (7-5). In this case, the limit imposed by the homogeneity constraint (5-66 on page 115) is not approached, and the model forms a more sensitive description of the real spectrum.

We constructed scatter matrices to represent misclassification errors made by both types of sampled pixel classifier with uncorrupted images (data not shown). No bias was observed, but the number of errors was so small that it was difficult to draw firm conclusions. Smooth-sampling reduces aliasing error and provides better protection against noise. It is unclear whether modelling improvements derive mainly from a reduction in the contrast of spectral peaks or an increase in peak width.

7.4.3 Effects of Sampling Blurred Images

Blur attenuates high spatial-frequencies, reducing the amount of aliasing that occurs when the image is sampled. For large amounts of blur, the image is effectively band-pass filtered, so that subsequent sub-sampling does not cause much loss of information. In this case, the net effect of blur and sampling is to apply a lower amount of blur to the sub-sampled image, reduced by the same factor as the image dimensions. Sub-sampling does not change the fact that blur distorts the spectrum, but has reduced the effective amount of blur. Our prediction that misclassification errors would favour models with a lower variance is still valid (§7.2.2). After the 33 image classes have been

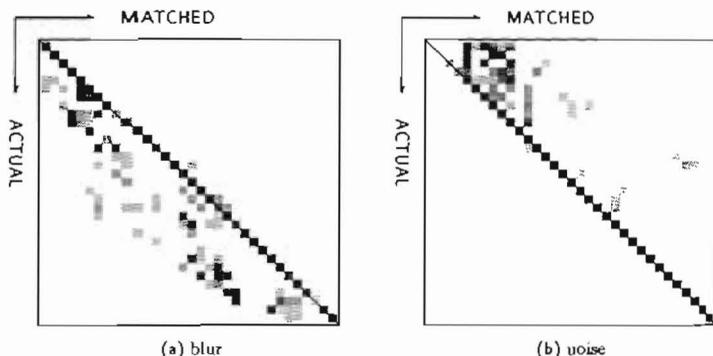


FIGURE 7.22: Scatter matrices for Sampled-Markov classifiers with corruption. Our Brodatz image set was classified using true-likelihood parameter sets by the All-Quarters method after sampling with a period of two pixels (§6.4.3). The matrices have been ordered by increasing MRF variance (33 classes). Correctly classified images lie along the leading diagonal, and intensity is proportional to frequency. (a) smooth-sampled classifier with noise, variance 100 gray-level units squared, orders 2, 5, 7 and 9; (b) sub-sampled classifier with blur, parameter 0.5 pixels, orders 3, 5 and 9. High levels of corruption have less effect than for the unsampled pixel classifier (Figures 7.15 and 7.17) but the trends towards higher variance (noise) and lower variance (blur) remain.

sorted by MRF variance (Figure 7.16), misclassification errors of this type are marked below the leading diagonal of the scatter matrix, and this type of error predominates (Figure 7.22a). Compare these results with those for the unsampled pixel classifier (Figure 7.15): the trends are similar, but the magnitude of the effect is smaller after sampling, and reduces further when the sampling period is increased (data not shown).

7.4.4 Effects of Sampling Noisy Images

Sub-Sampled Images

The expected effect of uncorrelated additive white noise is to add a constant to the image power spectrum, and this remains the case after sub-sampling with integer period s because aliasing still gives a flat noise spectrum. Consequently, classification errors should now show a trend in favour of models with a high MRF variance (§7.3.2), and this effect is observed in practice. The critical ratio which determines the magnitude of this effect is: ρ^2/σ^2 (§7.3.2).

From this argument, it could be thought that sub-sampling would have no effect on noise-immunity because noise has a similar effect on the spectra of dense and sampled images. Examination of Figure 6.21 on page 170 disproves this view: the sub-sampled classifiers have greatly-increased noise tolerance. This is because the MRF variance of sub-sampled models tends to be much higher than for dense images (Figure 7.3), and

hence the magnitude of the noise has effectively been reduced by the ratio ρ^2/σ^2 , tending to be masked by the inherent variability of the pattern. Informally, the increased variance may be taken as a reflection of the loss of information caused by sampling: the MRF predictor \hat{x} is very accurate for a dense image, giving a low predictor-error variance σ^2 , but sampling inevitably reduces accuracy and causes the predictor-error variance to rise. Unlike unsampled models, the variance changes by only a small proportion for different MRF order, explaining why an interaction between noise level and model order was virtually absent from our results (§6.6; data not shown).

Smooth-Sampled Images

On first glance, the smooth-sampled classifier has a big advantage over other pixel classifiers because the window function reduces the variance of the additive noise in the sampled image by a factor f^2 :

$$\begin{aligned} \text{var}\left(f^2 \sum_{i \in \mathcal{N}} \nu_i\right) &= f^4 \sum_{i \in \mathcal{N}} \text{var} \nu_i \\ &= f^2 \rho^2 \end{aligned} \quad (7.25)$$

where f is the sampling frequency, and the range of summation for i extends over the window neighbourhood \mathcal{N} . A few moments consideration show that the variance of the field is liable to be reduced by the same factor, nullifying this advantage. In practice, this does not happen because the field is usually dominated by low frequencies, which are hardly affected by the windowing function. By contrast, the noise is proportionately greatest at precisely the higher spatial frequencies which are most heavily attenuated, and its effect is reduced accordingly. Smooth-sampling with a period of 2 pixels reduces the average model variance by a factor 1.8 (Figure 7.3), but the variance of additive noise by a factor 4, leading to a two-fold increase in signal-to-noise ratio. For heavy sampling (small f), even large amounts of additive noise hardly affect classifier accuracy (§6.6.2; Figure 6.24 on page 172), but significant loss of information occurs at high levels of sampling and the error rate for uncorrupted images starts to rise. Despite this, the error rate for no added noise was still lower than for the dense classifier for the range of values we examined (§6.6.2). Scatter matrices confirm the expected prejudice for texture classes with high MRF variance (Figure 7.22b).

7.4.5 Improved Robustness: Modified Classifiers

Although the noise and blur-tolerance of the sub-sampled classifiers is much greater than with the dense image, it is possible to improve it further by using a modified classifier, similar to those suggested above (§7.2.3; §7.3.3). Frequency aliasing complicates this approach because it is not possible to predict what proportion of the spectral energy is caused by this effect. Aliasing may be ignored, effectively assuming that the original image was band-limited and that all energy observed in the sampled spectrum is genuine. The success of this approximation clearly depends on the particular image. It allows the previous blur-tolerant classifier to be used without further modification (§7.2.3), because the shape of the blur kernel is unchanged.

Additive noise presents less of a problem because its (expected) effect on the spectrum is known, and it is straightforward to correct for the effect of the window function in the case of the smooth-sampled classifier. Whether the noise-tolerant classifier generates useful results depends on the validity of our simple noise model, and should be verified experimentally. Neither of these approaches has currently been implemented.

7.4.6 Sampled Gabor–Markov Models

Convolution with a Gabor kernel performs a windowed transform, effectively blurring the image with a space-constant σ_{Gabor} (§4.1). The filter parameters we used imply that $\sigma_{\text{Gabor}} = 2.25$ pixels (§6.4.2), and hence it is not surprising that agreement between the unsampled data and auto-normal model is poor, given that a much smaller blur parameter, $b = 0.5$ pixels, was sufficient to cause difficulty for the pixel classifier (§6.5.3). High spatial-frequencies are heavily reduced by this amount of blur, creating a spectral peak of very high contrast. Figure 7.20 shows how close to zero the average minimum spectral density coefficient λ_{min} becomes for the composite-feature model, leading to higher spectral contrast than for pixel models. Only after a greater degree of sampling does the peak contrast decline to more familiar levels. The band-limited nature of the spectrum means that parameter sets are difficult to estimate and have a strong risk of numerical instability (page 196), but implies that very little information is lost on sub-sampling. Note that feature arrays are sampled for Gabor–Markov models rather than the original image (§6.4).

Blur inherent in the Gabor kernel tends to dominate the power spectrum, meaning that there is little to distinguish the shape of spectra from different textures, and the magnitude of the response and the mean of the feature array assume greater importance. Including the feature mean leads to a significant improvement in classification accuracy (§6.5.1; Figure 6.13 on page 166), and this trend accelerates as the sampling period is increased. When the feature mean is included, the error rate is still falling for the composite-feature classifier at a sampling period of four pixels whereas the rate for the pixel classifier has then started to rise (§6.6). This is perhaps explained by the peak in the feature spectrum being steeper, reducing the effect of aliasing (compare Figures 7.19 and 7.23).

7.5 Conclusions: Modelling Accuracy Determines Classifier Performance

Texture classification benchmarks show that our hybrid Gabor–Markov models perform better than a conventional pixel analysis, achieving superior robustness to image blur and noise combined with greater computational efficiency. Our proposed Sampled-Markov models have even more attractive properties, and achieved the best results in our tests (Chapter 6). In this Chapter, we examined the level of agreement between the imposed auto-normal model and experimental data, and found this correlated strongly with classifier performance. Unsampled pixel data was in moderate agreement with the model, and we identified image blur as a possible cause of modelling error. Synthetic

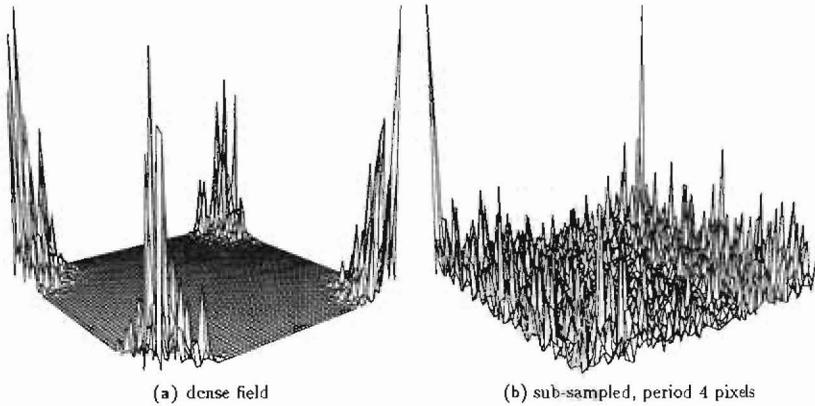


FIGURE 7.23: Power spectra of dense and sampled Gabor-Markov features. Horizontal axes represent spatial frequency, with the origin at any corner, and height represents image energy. Resultant features were extracted from the pressed-cork image (#4; Figure 1.6). (a) power spectrum of the composite-feature model: energy is concentrated at low spatial frequencies. (b) as (a), after sampling with a period 4 pixels: the spectrum is more even, but without the very high degree of aliasing seen for sampled pixel spectra (Figure 7.19).

textures fit the model exactly, but agreement soon deteriorates when the image is corrupted by noise or blur, consistent with poor robustness of the unsampled pixel classifier. By contrast, sampled pixel models were in close correspondence with the data, and this fit declined only gradually when the image was corrupted. Unsampled Gabor-Markov features show poor correspondence with the auto-normal model, and although this was substantially improved by sampling, skew was a problem for angular features. We proposed a modification to overcome this, forming improved Gabor-Markov models, and demonstrated that these were in significantly closer correspondence with the auto-normal model. From these observations, we conclude that classification accuracy is strongly correlated with the fidelity of statistical modelling. It is not sufficient to rely on the statistical framework to make best use of the data: pre-processing to ensure high descriptive accuracy affects performance considerably.

By examining the effects of image corruption on the true-likelihood function, we predicted trends in misclassification error arising from noise and blur, and demonstrated that our results followed this pattern. Blur biases the classifier towards models of lower MRF variance; noise towards higher variance. In both cases, we suggested modified classifiers having the flexibility to resist these predispositions, and demonstrated that classification accuracy could be improved substantially by this approach.

Images influence the likelihood function only through their power spectrum, and we sought to explain the consequences of image sampling by a spatial-frequency analysis. We suggested that the presence of a high-contrast peak in the image spectrum causes modelling difficulty, and hence leads to poor classifier performance. Tall peaks usually

occur in the power spectra of our Brodatz textures, but peak contrast is reduced by sampling, leading to better agreement with the model and improved classifier accuracy. This conjecture should be understood as a heuristic argument supported by experimental data rather than a proof. When taken to extremes, sampling produces data which may be modelled very accurately but contains little remaining information. This limit was not approached by the parameters used in Chapter 6, and our results show that performance of all types of classifier correlates closely with the level of correspondence with the auto-normal model.

8

Conclusions, and Suggestions for Further Work

This thesis appertains principally to the efficient analysis of visual texture. Blending the computational efficiency of Gabor filtering with the descriptive power of hierarchical Gibbs–Markov models, we have proposed a statistical Gabor–Markov framework for texture analysis in low-level vision. Our experimental appraisal of this framework led us to propose families of Sampled-Markov models. Adopting texture classification accuracy as a benchmark, we showed that both model types combine improved efficiency with greater accuracy and robustness than a conventional Markovian analysis. These performance trends were explained by examining interactions between image data and the auto-normal Markov model: there is a strong correlation between accuracy of classification and fidelity of the statistical modelling.

In this final Chapter, we restate our main objectives, summarise how we have sought to meet them, and draw conclusions from our achievements. We suggest several avenues for further research arising from extensions and refinements of our work. Gabor–Markov models are potentially more flexible than conventional Markovian approaches to texture, but this was not exploited fully by the prototype models we employed. Enhancing the range of features and image primitives coupled with automatic spatial-frequency tuning should remedy this omission. Image sampling lies at the heart of our most successful Sampled-Markov texture classifiers, and we propose investigating the influence of anti-aliasing to test whether their superiority may be further increased. Finally, we mention ideas for additional texture classification and segmentation experiments, and comment on approaches to the implementation of Gabor–Markov models on parallel hardware.

8.1 Markovian Analysis of Texture: Conclusions

Subjectively, human visual perception appears to be immediate and spontaneous, allowing us to sense our surroundings completely and effortlessly, but controlled experiments demonstrate the existence of both serial and parallel paradigms (Chapter 2). Current psychological models of low-level human vision employ a two-stage hierarchy: simple spatially-parallel “pre-attentive” filtering guides the allocation of more complex but spatially-restricted “attentive” processing across the visual field. Similarly in machine vision, it is possible to analyse texture either in terms of derived properties or statistics, many of which may be obtained by simple local processing; or in greater depth but with more difficulty, by characterising its hierarchical structure (Chapter 3). This thesis is about serial and parallel paradigms in low-level vision.

Derived texture statistics are usually incomplete, and hence fail to represent each texture unambiguously, but are computationally attractive because local data-independent processing maps easily onto special-purpose parallel hardware (in both biological and machine vision). We have examined Gabor filters in this thesis because of their unique theoretical advantages for signal processing, namely minimisation of joint uncertainty according to the uncertainty principle for information (Chapter 4). Despite these attractive properties of Gabor kernels, simple image segmentation algorithms derived from idealised texture models gave mixed results. Suitably-constrained textures were segmented accurately and efficiently, attaining levels of performance qualitatively similar to pre-attentive human vision. Performance deteriorated for textures violating the assumed model, and borders were detected with low accuracy or missed altogether.

Natural textures often display random rather than regular spatial variation, and may possess different structure at different spatial scales (Ahuja & Rosenfeld, 1981; Marr, 1976). Notions of optimality, hypothesis-testing and inherent variability are central to statistical study, and we adopt Gibbs-Markov random field models because they make explicit the influence of local spatial context within a statistical framework (Chapter 5). Raw random field models operate at a low level, describing arrangements of individual pixels, and it is common to describe more complex texture structure by hierarchical models. Unfortunately, algorithms to manipulate these structural models are often computationally intense, and several approximations must be taken to simplify the analysis, diminishing the benefits arising from an optimal statistical framework. Gibbs-Markov models have been successfully applied to many classic image-processing problems, including image restoration and unsupervised segmentation.

8.1.1 Our Approach and Objectives

The chief objective of this thesis is to propose new models for texture analysis combining the computational efficiency of Gabor filtering with the superior descriptive power and optimal statistical framework available to hierarchical Gibbs-Markov models. Limitations of current Gabor-filtering algorithms arise from over-simplified image models: textures are assumed to contain few significant spectral components and to generate constant or smoothly-varying Gabor amplitude when filtered (Bovik *et al.*, 1990; Tan, 1988). In practice, the spectral composition of adjacent textures may not be disjoint,

and Gabor amplitude shows considerable variation *within* each textured region (Chapter 4). Some authors have sought to suppress these fluctuations by employing heuristic post-processing, but this leads to distorted or missing texture boundaries (*e.g.* Fogel & Sagi, 1989). Our approach is to *exploit* rather than suppress spatial variation in Gabor amplitude within an optimal statistical framework, describing the spatial arrangement of Gabor “features” with Markov random fields to form hierarchical *Gabor-Markov* models.

Gabor feature arrays are expressed in standard forms in order to hide details of low-level filtering from higher-level models, and to provide a degree of abstraction (Chapter 4). Unlike previous expedient strategies for combining measurements from many Gabor channels, we propose two feature-extraction methods based on sound principles. Our Profile algorithms determine local image features by matching observed channel responses to ideal Gabor signatures generated by parameterised image primitives, selecting the feature vector that brings these into closest correspondence. In an alternative procedure, Resultant features are obtained directly from Gabor channel outputs by exploiting response moments. Both methods generate sensitive and compact descriptions of real textures, and may be enhanced by the improvements suggested below (§8.2).

Markov random fields are specified by local conditional probability densities arising from pixel interactions within a small image neighbourhood (Chapter 5). Hypothesis-testing and manipulation of these models often requires the *joint* probability of *all* pixels to be computed, but this is intractable except in special cases. We adopt auto-normal Markov models for this reason, assuming that pixels have a normal marginal probability density and interact linearly. Orientation is an important visual cue and a key Gabor feature, but is periodic whereas the normal distribution is aperiodic and unbounded. We proposed angular field models to overcome this disparity.

In order to compare the performance of our proposed Gabor-Markov models with a conventional Markovian analysis, we adopted texture-classification accuracy as a benchmark (Chapter 6). True and pseudo-likelihood classifiers were implemented for both paradigms, and we tested their robustness by adopting ideal models for image noise and blur. Fixed filter parameters were used to extract Gabor features from each image, but automatic spatial frequency-tuning methods proposed below should lead to more sensitive and flexible representations (§8.2).

8.1.2 Experimental Results and Discussion

Similar accuracy was obtained by Gabor-Markov and conventional pixel classifiers with uncorrupted images, achieving a minimum error rate of 6% (Chapter 6). Our Gabor-Markov analysis was much more robust, increasing noise-tolerance by a factor of up to 45, and derives computational benefits from a reduction of effective image area of up to 94%. Little distinction was observed between performance with Profile and Resultant feature vectors, but true-likelihood parameter sets were consistently superior to pseudo likelihood. Further investigation led to the discovery of Sampled-Markov classifiers: these enjoy greater robustness and higher computational efficiency than both conventional and Gabor-Markov classifiers, and have an optimal error rate of 0%.

Although we expected Gabor–Markov classifiers to perform well, the success of Sampled–Markov classifiers is remarkable. Classification accuracy shows strong correlation with the level of correspondence between image data and the assumed auto-normal model, and our results may be explained by this effect (Chapter 7). Parameter sets for conventional pixel models show poor agreement with the image data from which they were estimated, possibly due to image blur. By contrast, Sampled- and Gabor–Markov models both fit experimental data closely, particularly after proposed improvements (§7.1.5; §8.3). We believe that the presence of high-contrast peaks in the image power spectrum leads to inaccurate modelling by a conventional analysis, and demonstrated that peak contrast is reduced by sampling, but further investigation of this effect is proposed below (§8.3.1). Trends in misclassification error are predicted correctly by examining the interaction of image blur and noise terms with the auto-normal likelihood function, and modified classifiers were proposed to compensate for these types of image degradation.

8.1.3 Summary of Achievements

In this thesis, we have:

- explored the applicability of Gabor energy filtering to visual texture segmentation, and identified causes for observed performance limitations. Variability is often observed within each filtered texture, but this is not accommodated by algorithms responding to first-order differences of Gabor amplitude.
- devised sensitive and efficient algorithms to extract Gabor features from textured images, and proposed a hybrid Gabor–Markov framework for texture analysis.
- discovered new families of Sampled–Markov texture models possessing attractive descriptive and computational attributes.
- adopted texture classification accuracy as a benchmark to compare our novel frameworks with a conventional Markovian analysis.

Gabor–Markov: without reducing classifier accuracy, computational load was reduced by 88%, and robustness to image noise increased by a factor of up to 45.

Sampled–Markov: computational load was reduced by 96% and robustness to image noise increased by a factor of up to 200. Smooth–Sampled models achieved 100% accuracy in our tests, compared with 94% with a conventional analysis, and are the preferred choice.

- explained observed performance trends by investigating interactions with the assumed auto-normal Markov model. Classification accuracy correlates strongly with the descriptive accuracy our models, establishing the importance of appropriate pre-processing of image data prior to statistical analysis.

Our classification results demonstrate the superiority of both Sampled–Markov and Gabor–Markov models compared to a conventional Markovian analysis. Accordingly,

we conclude that the conventional approach should be discarded except for analysis of very small image regions, when there is insufficient data for Sampled- or Gabor-Markov models to operate reliably. Further work suggested below can only widen the performance gap between our proposed models and current methods, strengthening this conclusion. Highest classification accuracy with our Brodatz image set was achieved with a fifth-order true-likelihood Smooth-Sampled Markov classifier, sampling period 2 pixels, and we recommend these model parameters. Gabor-Markov approaches are potentially more flexible than Sampled-Markov models, and could become the preferred paradigm if the improvements and further work suggested below are successful.

8.2 Improved Gabor Feature Representations

Our novel Gabor-Markov models achieved superior performance to a conventional pixel analysis in the texture classification benchmarks reported in Chapter 6, suggesting that textures are characterised adequately by arrays of Gabor features extracted by our proposed methods (§4.6). Despite this success, several aspects of the feature-extraction process could be improved, leading to increased flexibility, better performance, and greater computational efficiency. With these enhancements, Gabor-Markov models may surpass the excellent results obtained with Sampled-Markov models.

8.2.1 Additional Features and Image Primitives

Our proposed Profile feature-extraction method combines measurements from many Gabor channels by matching observed responses to Gabor signatures produced by parameterised image primitives (§4.6.2). This procedure is flexible because additional Gabor channels or alternative filter parameterisations may be accommodated easily, yet the representation always assumes a standard form. Such an arrangement is convenient for a hierarchical model, as it hides low-level operational details from the high-level model (our Markov random fields). We noted in §4.5.2 that the success of this approach is conditional on the selection of appropriate image primitives and parameterisations. Two primitives were discussed above, LINE and SINE, corresponding to an ideal line and a sinusoid grating, respectively. Other primitives may provide a more appropriate description of the observed Gabor responses in some cases, and we have considered STEP and RAMP, corresponding to a step-edge and intensity ramp:

$$f_{\text{STEP}}(x, y) = \begin{cases} c & \text{if } x < 0 \\ -c & \text{if } x > 0 \end{cases} \quad (8-1)$$

$$f_{\text{RAMP}}(x, y) = cx \quad (8-2)$$

where both primitives may be rotated about the origin, and possibly translated. With an extended range of image primitives, it may be appropriate to select the signature which most-closely matches the observed response at each image location, perhaps using “primitive type” as an additional image feature.

No image primitives are required for Resultant feature extraction: observed Gabor responses are described directly (§4.6.3). Similar performance was obtained with Profile

and Resultant features in conjunction with Gabor–Markov texture classifiers (§6.5), but this was using filters tuned to a single spatial frequency. Two deficiencies of Resultant feature extraction may be addressed by a simple extension to the procedures described in §4.6, by estimating a local Gabor “phase” feature, $\hat{\phi}$. Firstly, the current method does not measure the *sign* of local contrast c , because both contrast and orientation features are extracted from measured Gabor energies, which are insensitive to this parameter ($\mathcal{R} \propto c^2$). This is probably the cause of skew present in direct MRF models of image contrast (e.g. Figure 7.10*b* on page 189). Secondly, it is not possible at present to reconstruct textures from their Gabor features because these do not capture sufficient information. Texture synthesis is desirable to verify that feature sets represent visual properties of the texture adequately, and often suggests how improvements may be made (§3.6.1). We could, for example, compare textures synthesised from their Gabor and MRF parameter sets, and ultimately also from Gabor–Markov parameters. Approximate reconstruction is possible if local phase is known, and phase also determines the sign of local image contrast.

We have tentatively defined our Gabor “phase” feature $\hat{\phi}$ in a similar manner to other Resultant features:

$$\tan \hat{\phi} = \frac{\sum_i M_{s,i}}{\sum_i M_{c,i}} \quad (8.3)$$

where summation for i extends over all channels, M_s and M_c are the sine and cosine phases of observed channel responses (orientation spacing assumed equal), and $-\pi < \hat{\phi} \leq \pi$. Examples of phase fields extracted from two of the images used previously are shown in Figure 8.1: whilst we have not attempted to classify textures based on these features, they do appear to have potential for image description, and phase discontinuities may be used to locate texture boundaries (Bovik *et al.* 1990). Phase could easily be added to the Gabor–Markov models described in Chapter 6, either as an “independent” additional feature, or by modifying an existing model, perhaps forming a complex-valued Markov random field.

8.2.2 Selection of Gabor Filter Parameters

Fixed Gabor filter parameters were used to extract features used by the Gabor–Markov classifiers reported in Chapter 6: each image was filtered at six orientations, two phases, and a single spatial frequency (§6.4.2). Natural textures often have a hierarchical (rather than a fractal) structure, and may generate quite different feature arrays at different spatial scales (Ahuja & Rosenfeld, 1981; Marr, 1976). Both our feature-extraction methods, Profile and Resultant, could easily be extended to accommodate variable or multiple spatial-frequency tuning, allowing the higher-level Markov random field component of our hierarchical models to operate at a more abstract level. Our Gabor segmentation experiments confirmed that fixed filter parameters are not always suitable for texture discrimination (§4.4), and we suggest two alternative procedures below.

Automatic Selection of Spatial-Frequency Parameter

When the objective is to represent the most significant component of a texture, or to characterise it approximately using only a few filter channels, the power-spectrum

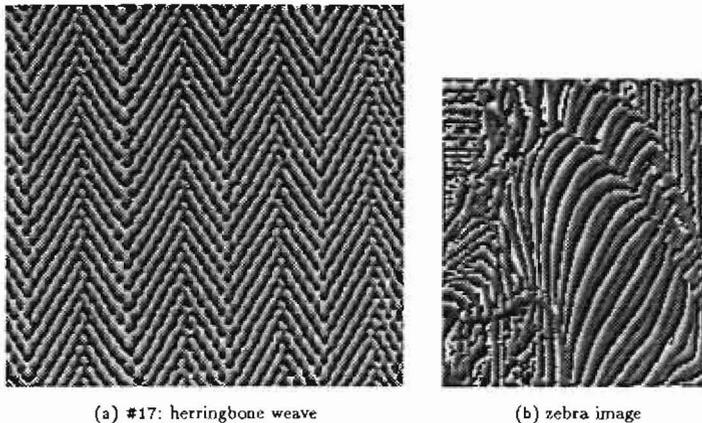


FIGURE 8.1: Examples of proposed Gabor “phase” feature. Gabor “phase” feature ϕ drawn to a linear scale (white: $\phi = -\pi$; black: $\phi = \pi$). (a) herringbone image (#17; Figure 1.6 on page 11), 256×256 pixels, filter parameters: ($T = 7.3$ pixels, $B_{1/2} = 0.7$, $\Delta\theta_{1/2} = \pi/9$, $\Theta = \pi/6$; $\sigma = 5.8$ pixels). (b) zebra image (Figure 4.4 on page 68), 151×177 pixels, filter parameters: ($T = 6$ pixels, $B_{1/2} = 4$, $\Delta\theta_{1/2} = 2\pi/9$, $\Theta = \pi/4$; $\sigma = 1.3$ pixels), using the notation described on page 69. Results with these images suggest that the proposed phase feature generates useful image descriptions.

method is often able to suggest suitable Gabor filter parameters (Bovik *et al.*, 1990; Tan, 1988). Dominant spatial frequencies are found by searching for peaks in the image power spectrum (§4.3.2). We adopted the power-spectrum method to select filter parameters used to obtain Gabor energy segmentations (Figures 4.14 and 4.15 on pages 78–80) and Gabor-feature representations (Figures 4.24–4.26 on pages 96–98), described in Chapter 4. In the form stated in §4.3.2, the power-spectrum method determines both a spatial frequency and an orientation by locating a single spectral peak. We intend to sample image orientations evenly at the chosen spatial frequency, and hence it is more appropriate to pool spectral energy across all orientations before locating frequency co-ordinates of spectral peaks.

When the application is texture classification, and projection parameters may be assumed constant across the image, a single spatial scale chosen by the power-spectrum method will probably give a superior feature representation than a fixed value assumed *a priori*. Feature representations may be made scale-invariant by sampling in proportion to the filter period, hiding changes in spatial scale from the random field components of our hybrid models. This is attractive because Markov random field models are sensitive to changes of texture magnification, and an extension by means of a modified classifier is awkward and computationally demanding (Cohen *et al.*, 1991). Elimination of scale variation during the Gabor filtering stage may be achieved with very little computational effort, using the method suggested above, which also automatically determines the sampling period for Gabor–Markov models. Care is required when competing frequencies have similar energies because selecting different bands during training and

classification stages could prove disastrous. Unlike true modified Bayes classifiers, this method has no claim to be optimal, but is more computationally efficient.

Raw MRF models are also sensitive to image rotation, unless circularly-symmetric features are used (§3.1.3). Image rotation is effectively eliminated by the improved Gabor-Markov models proposed in §7.1.5 because the orientation feature array is effectively "normalised" before computing fields required by the Markov models. Both spatial-frequency and orientation parameters may be estimated automatically with little computational effort, providing (approximate) scale- and rotation-invariant texture classification more readily than full modified Bayes classification (Cohen *et al.*, 1991).

Filtering at Multiple Spatial Scales

Only trivial textures possess significant structure at just one spatial scale, and more sensitive feature representations to those considered above require filtering at multiple spatial-frequencies. Suitable frequencies may be identified by locating peaks of successive magnitude in the orientation-corrected power spectrum (Tan, 1988), or a fixed range of frequencies may be built into the feature-extraction algorithm. With this extra flexibility, it is possible to compute a *local* scale parameter, perhaps along the lines described in §A.3. Our signature-matching (Profile) approach to feature extraction adapts immediately to channels tuned to many spatial frequencies, allowing us to estimate more feature parameters for our current image primitives, or to consider more complex patterns.

Adopting a local spatial scale is appropriate for image segmentation, where different textures (or portions of textures) may be present at different magnification. The "scale" parameter could be treated as another "independent" feature and added to existing Gabor-Markov models, or possibly used to obtain an initial rough segmentation so that later MRF algorithms could be applied at a single scale. A further possibility is to apply the information represented by the scale feature to estimating slant and tilt, neither of which is accommodated well by raw Markov models.

We have demonstrated a link between classifier performance and the level of correspondence between texture model and experimental data (Chapter 7), and showed that improved Gabor-Markov models achieve good agreement (§7.1.5). Coupled with improvements in feature representations and filter tuning proposed here, we believe that Gabor-Markov models are capable of higher accuracy and greater flexibility than the prototype versions implemented in Chapter 6, and are deserving of further study.

8.3 Improved MRF Models

Our experience with unsampled MRF pixel models has shown that the use of maximum-likelihood techniques does not necessarily lead to an optimal system (Chapter 6). We have a number of ideas for improvements and suggestions for further work, mostly stemming from the success of Sampled-Markov models.

8.3.1 Further Investigation of the Effects of Sampling

Our examination of the influence of image sampling on classifier performance found convincing explanations for the superior blur and noise tolerance of Sampled-Markov texture models (§7.4), and we found experimentally that performance correlates closely with the level of correspondence between model and data (§7.1). We suggested that unsampled arrays of pixels (and Gabor features) are difficult to model closely by auto-normal MRFs because of the presence of high-contrast peaks in their Fourier spectrum, supporting this hypothesis with experimental data (§7.4). The practical importance of the consequences of image sampling justify further investigation of this topic. For example, might it be possible to determine the most appropriate sampling period automatically by examining the image power spectrum? This procedure would have further practical advantages, allowing the most favourable trade-off between modelling accuracy and computational efficiency to be determined for each texture class.

The role of frequency aliasing in the superior performance of Sampled-Markov classifiers has not been fully examined. Aliasing contributes to an increase in high-frequency energy in sampled images, acting to reduce the contrast of the problematic spectral peak, and hence may be beneficial in small amounts. This suggestion is not entirely consistent with the success of Smooth-Sampled Markov classifiers, however, since their window function reduces aliasing. Anti-aliasing is computationally-expensive in the spatial domain, requiring convolution with a broad kernel, but may be performed easily with an unsampled spectral representation by band-limiting it. Adopting this procedure could allow us to eliminate aliasing error, and hence achieve a better understanding of the effects of image sampling, and possibly obtain yet further improvements in classifier performance. Manipulating image dimensions in the spatial-frequency domain also facilitates the use of non-integer sampling periods.

Preliminary Investigation of the Influence of Anti-Aliasing

Thorough investigation of the influence of frequency aliasing awaits further research, but we may report some preliminary results with the pressed-cork image (#4; Figure 1.6).¹ The image was band-limited in the Fourier domain, and converted back to its spatial form. We characterise anti-aliasing by the largest sampling period with which the modified image may be sampled without introducing aliasing error. Band-limited versions of the pressed-cork image are shown in Figure 8.2: the visual effect of anti-aliasing is similar to blur. We may compare power spectra of sampled band-limited images, without aliasing, with those of the unprocessed sampled images (Figures 8.3a,b and 7.19b,d on page 207, respectively), shown from the same viewpoint. Power spectra with a sampling period of 2 pixels are similar, but the anti-aliased version assumes a much less chaotic form for heavier sampling (period 4 pixels). This confirms that the original image is virtually band-limited, and hence that aliasing plays a minor role for moderate sampling, becoming more significant when the sampling period is increased. Spectral peaks of band-limited images have very high contrast, tending to infinity, and hence

¹Unlike previous Chapters, where results with the pressed-cork image have been presented to illustrate common trends for our Brodatz image set, the investigation presented here is preliminary and we have only examined this image.

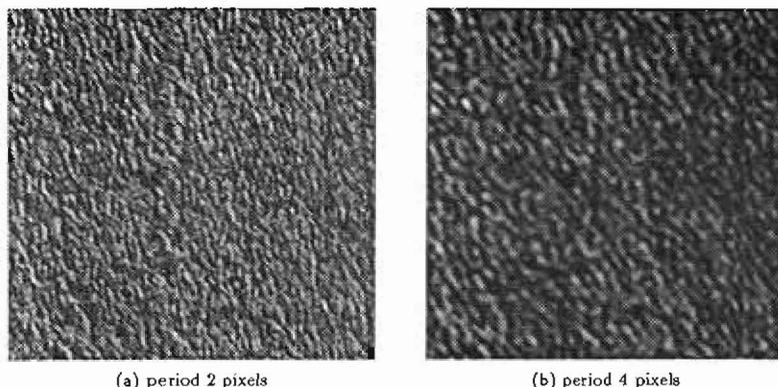


FIGURE 8.2: **Anti-aliased Brodatz image.** Anti-aliased versions of the pressed-cork image (#4; Figure 1.6 on page 11), formed by adjusting its Fourier spectrum. (a) suitable for sampling with period 2 pixels; (b) period 4 pixels. Anti-aliasing has a similar visual effect to blur.

our argument of §7.4 predicts that modelling error will be high. Agreement with ninth-order true-likelihood MRF parameter sets after anti-aliasing for sampling periods of 2 and 4 pixels are shown in Figure 8.3c,d: modelling error is very significant, appearing similar to that for blurred images (§7.1). Note the extremely low standard deviation of the empirical MRF predictor-error, 0.05 and 0.06 gray-level units, respectively: centre pixels of band-limited images may be reconstructed almost exactly from their MRF neighbourhoods.

By contrast, correspondence between ninth-order true-likelihood parameter sets estimated from sampled anti-aliased pressed-cork images is very close (Figure 8.3e,f). The degree of anti-aliasing matches the sampling period, so that the resulting frequency spectrum does not contain any “holes” (a,b), and contrast of the spectral peaks does not approach infinity. From these preliminary results, which have been obtained by processing only the pressed-cork image, it appears that frequency aliasing is not beneficial to modelling accuracy, and should be eliminated by anti-aliasing prior to sampling. Once this is done, we anticipate that performance of anti-aliased sampled classifiers will be superior to smooth-sampling (§6.6.2), further extending the advantages of our proposed methods compared to conventional techniques. Sampled Gabor-Markov models would probably also benefit from anti-aliasing, but this was not examined. If this tentative conclusion is supported by further research, automatic selection of the most appropriate sampling period becomes even more attractive, corresponding to the proportion of the frequency spectrum to be modelled.

8.3.2 Simplified MRF Neighbourhood Structure

Classification results for both sampled and unsampled images show that there is little performance benefit when MRF neighbourhood size is increased above a small minimum

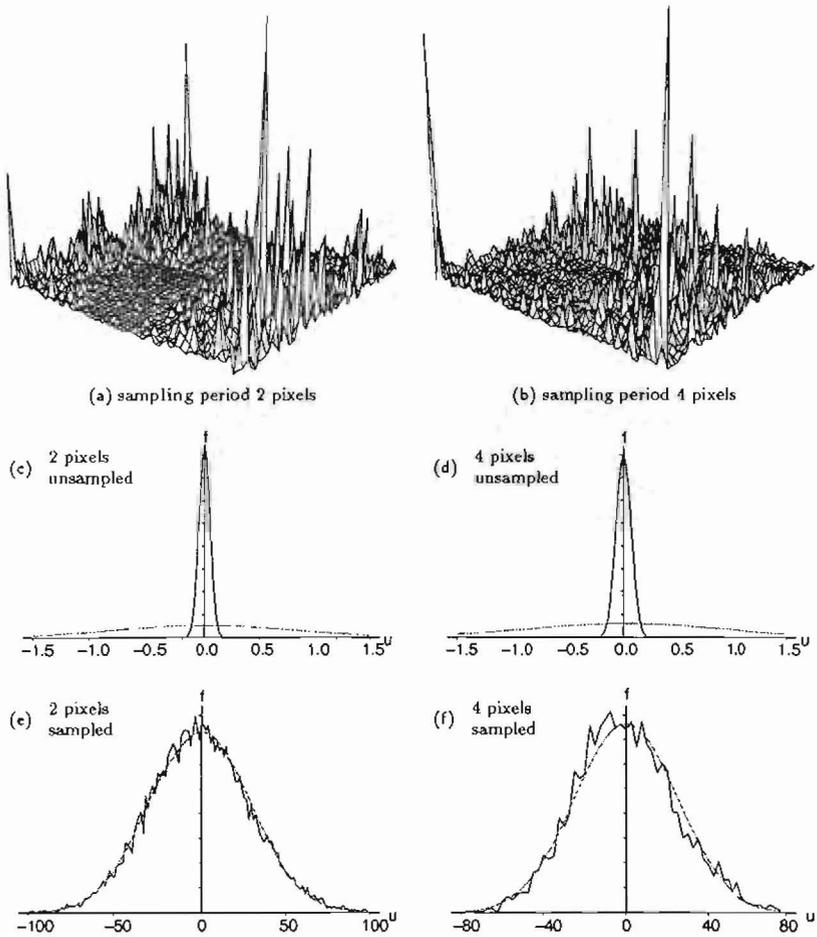


FIGURE 8.3: Effect of anti-aliasing on image spectra and correspondence with auto-normal model. The pressed-cork image (#4; Figure 1.6 on page 11) was anti-aliased by adjusting its Fourier transform to permit sampling with the period shown, and ninth-order true-likelihood parameter sets estimated. (a,b) power spectra after sampling, periods 2 and 4 pixels, respectively. Horizontal axes represent spatial frequency, with the origin at any corner. Compare spectra without aliasing (Figure 7.19b,d on page 207). (c,d) histograms of observed (solid) and expected (dotted) distributions for MRF predictor-error u (gray-level units), showing agreement with auto-normal model of unsampled band-limited image, periods 2 and 4 pixels: very poor correspondence. (e,f) after sampling, periods 2 and 4 pixels: close correspondence. Compare agreement with original image (Figure 7.1a on page 178) and after raw sampling (Figure 7.6a on page 187).

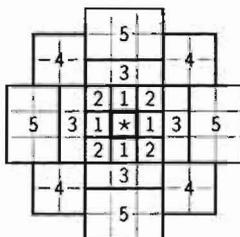


FIGURE 8.4: **Modified Markov neighbourhood structure.** This Markov random field structure has a larger image “footprint” than a conventional ninth-order model, but is specified by 10 rather than 24 parameters (compare Figure 5.1 on page 107).

value between third and fifth order (Chapter 6). In some cases, larger MRF neighbourhood size leads to *lower* accuracy! Third-order MRF neighbourhoods are specified by six parameters, rising to twenty-four for ninth order, and the additional computational effort involved in estimating and manipulating these extra terms hardly seems worthwhile. In some cases, it may be appropriate to adopt a simplified neighbourhood structure, where several pixels are multiplied by the same coefficient. The neighbourhood structure shown in Figure 8.4 has a larger image “footprint” than a ninth-order model, but is specified by only ten parameters. This form is drawn only to illustrate the idea; we have not performed experiments to determine its effect on performance. Grouping neighbourhood sites together in this way does not affect their linear interaction, and hence the resulting model is still auto-normal.

8.3.3 Experimental Appraisal of Modified Classifiers

We have characterised robustness with respect to idealised models for image degradation, modelling image blur by convolution with a Gaussian kernel, and image noise as independent, additive, white and Gaussian (§6.4.3). By examining the perturbing effect of these transformations on the image power spectrum, we identified *modified* Bayes classifiers which sense the amount of noise or blur present and correct appropriately (§7.2; §7.3). Preliminary results with the modified anti-blur classifier were encouraging (§7.2.3), but full investigation of the practical utility of these forms awaits further experimental verification. In particular, we need to establish whether real image degradation phenomena are modelled adequately by the idealised forms we have assumed, whether modified classifiers function correctly in conjunction with image sampling, and whether they are themselves sufficiently robust. Following a suggestion by Kashyap, we could design a further modified classifier insensitive to changes of global illumination (Kashyap *et al.*, 1982). Modified classifiers insensitive to global changes in image scale or rotation have been proposed by Cohen *et al.* (1991), and it would be interesting to compare the performance of these maximum-likelihood classifiers with our proposed sub-optimal method exploiting Gabor filtering (§8.2.2). All the classifiers mentioned above assume *global* image transformations; it would be useful to extend this to include smooth variations due to orthographic projection or texture mapping.

It has been suggested that manufactured objects (rather than natural textures of the type we have considered) are better described by non-stationary Markov models, where the field mean is a low-order function of image location (Hunt & Cannon, 1976; Silverman & Cooper, 1988). Parameter estimation and manipulation of non-stationary models are challenging because many of the techniques described above rely on the field being stationary (Chapter 5), and exact extensions to non-stationary fields are computationally prohibitive (Silverman & Cooper, 1988). Further development of efficient sub-optimal algorithms is required to make non-stationary models more accessible.

8.4 Applications to Image Analysis

Some applications of conventional Markov models to image analysis were discussed in Chapter 5, and many of these algorithms adapt easily to Sampled- or Gabor-Markov models. Texture analysis is an important part of low-level vision: there are very many tasks for which it may be exploited, and a full review lies outside the scope of this thesis. Some common examples include: classification of medical, aerial and geological images; navigation, bin-picking, and identification. In this section, we suggest some further texture classification experiments and comment on ideas for image segmentation algorithms.

8.4.1 Further Texture Classification Experiments

Texture classification was used in Chapter 6 as a benchmark to compare the performance of our proposed Gabor-Markov models with a conventional texture analysis. We concluded that Gabor-Markov models were superior to conventional unsampled pixel models with respect to classifier robustness and computational efficiency, and that the composite-feature model was usually the appropriate choice (§6.7). Sampled-Markov classifiers were much more accurate than other types, achieving 100% accuracy in some circumstances. Both Gabor-Markov and Sampled-Markov models achieve superior performance to that offered by a conventional analysis. Discussion of these results led us to suggest modifications to the original Sampled- and Gabor-Markov models, and further classification experiments suitable for establishing the merits of these updated proposals are considered below.

In §7.1.5 we proposed improvements to Gabor-Markov models, bringing them into closer correspondence with experimental data, and further enhancements were suggested above (§8.2). Our preliminary investigation of anti-aliasing suggests that Sampled-Markov models could also be improved (§8.3.1). Together, these proposed revisions should further widen the gap between conventional (unsampled pixel) analysis and the new forms we propose, but the margin should be determined experimentally. Our study has used natural Brodatz textures (Figure 1.6), and it would be helpful to extend the generality of our conclusions by repeating the classification exercise for a wider variety of real textures (aerial, medical, geological, *etc.*), perhaps also widening the domain to include colour images or range data.

Robustness to image noise and blur was measured in Chapter 6, and we have designed modified classifiers to increase immunity to these types of image corruption. In §8.2, we proposed that image scale and rotation should be eliminated from our hybrid models by adaptive Gabor filtering, believing that this procedure should be less computationally-demanding than full modified Bayes classification and demonstrate further advantages of Gabor–Markov models. Both proposals await experimental verification.

8.4.2 Ideas for Image Segmentation Algorithms

In this thesis, we have concentrated mainly on texture classification because it is a suitable benchmark for comparing different texture models, but image segmentation is probably of greater practical interest. Current approaches to image segmentation were reviewed briefly in §3.5, commenting further on Gabor and Markov methods in Chapters 4 and 5, respectively. Many of these conventional methods may be used in conjunction with our proposed Sampled- and Gabor–Markov models. Below, we comment on two ideas for efficient image segmentation, neither of which has yet been implemented.

Gabor–Gibbs Image Segmentation

Texture energy algorithms are attractive because of their computational efficiency, and may be implemented easily on parallel hardware. Our experiments with Gabor energy segmentation confirmed impressions from the literature that the main drawback with this approach is the level of variation within textured regions after filtering, so that boundary shape is often distorted by corrective post-processing (§4.4). Our approach to this situation is to describe the spatial arrangement of Gabor features by Markov random fields, forming Gabor–Markov models (§5.7). Gabor filter output may be sub-sampled, and hence these models retain some of the computational benefits of Gabor filtering.

An alternative procedure, more in keeping with the attentive–pre-attentive dichotomy described in Chapter 2, is to “regularise” Gabor output by modelling the segmented image with a Gibbs random field. Previous applications of Gibbs random fields (GRFs) to image segmentation have either used pixel data directly or formed a hierarchical random field model, and demonstrate that it is straightforward to assume suitable GRF clique potentials without having to estimate them explicitly (Chapter 5). The simple “modal filtering” algorithm we used above in conjunction with Gabor energy classifies each pixel according to the orientation of the dominant filter response, and is not very discerning (§4.4). A suitable extension of this approach is to form the Gabor orientation field using our proposed extraction method (§4.6), and then to “restore” it with the Gibbs sampling algorithm (§5.3.2; Geman & Geman, 1984). Raw feature arrays are “noisy” but seem amenable to improvement by this approach (Figure 8.5). Definitions for the clique potentials and image penalty term should reflect the periodicity of the data; perhaps something similar to $f_{\theta}(\theta)$ (Figure A.7 on page 268). Extension to other Gabor features, or to include additional boundary processes, is straightforward.

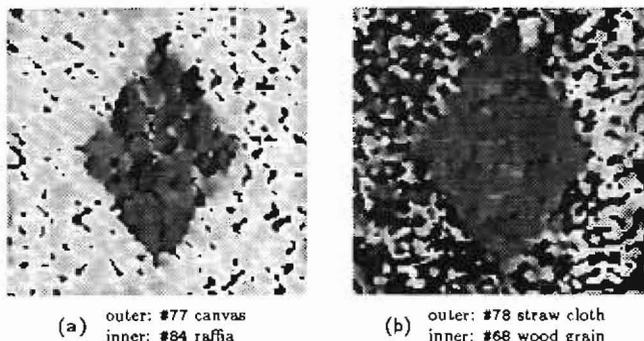


FIGURE 8.5: Gabor features suitable for modelling with Gibbs random fields. Orientation features estimated from Brodatz montages after filtering with parameters suggested by the power-spectrum method (§4.3.2). Orientation is represented by a linear scale from white to black, ignoring periodicity. (a) canvas-raffia (Figure 4.14*b* on page 78), filter parameters: ($T = 4.5$ pixels, $B_{1/2} = 0.7$, $\Delta\theta_{1/2} = \pi/6$, $\Theta = \pi/4$; $\sigma = 3.5$ pixels); (b) cloth-wood (Figure 4.14*e*), parameters: ($T = 5$ pixels, $B_{1/2} = 1.5$, $\Delta\theta_{1/2} = 5\pi/36$, $\Theta = \pi/6$; $\sigma = 2$ pixels), using the notation described on page 69. Both feature arrays are “noisy” but could probably be restored more sensitively after modelling by a Gibbs random field than by blur.

Essentially, the above proposal parallels models of human vision, using fast spatially-parallel Gabor filtering to guide more detailed examination of boundary location (§2.4.1). Gibbs random fields are suitable for modelling image layout because segmentations differing with respect to local change in boundary location may be compared by a local analysis. An alternative application is to use the primitive Gabor segmentation to guide estimation of initial MRF parameter sets in an estimate-segment cycle for unsupervised image segmentation (§5.6).

Sampled-Markov Image Segmentation Algorithm

Images usually have a very large number of possible segmentations, and computationally-efficient sub-optimal techniques must be employed to identify good solutions. A common approach is to exploit the phenomenon of spatial coherence (§2.3.4 and §3.5). In §5.6.3, we commented on a two-stage MRF segmentation algorithm proposed by Cohen and Cooper (1987), in which successively smaller image regions are classified according to known texture models. Large-scale image blocks serve only to determine the neighbourhood structure of smaller regions, but require significant computational effort to process because they each refer to a large amount of underlying pixel data. A very simple extension to this algorithm, exploiting the assumption that possible texture types are known *a priori*, is to match the sampling period to the dimensions of each image block, so that the volume of data accompanying each block is bounded. This allows coarse-scale segmentation to proceed rapidly, since few blocks are present, concentrating most computational effort on fine-scale refinement.

8.4.3 Implementation on Parallel Hardware

Even after efficiency gains accruing from our proposed improvements, image analysis with Markov models is computationally intense, stemming mainly from a requirement to evaluate relative image likelihoods for many possible scene configurations. Several of the application areas mentioned above demand real-time operation (e.g. navigation), and there is continual pressure to reduce execution time. An attractive route by which this may perhaps be accomplished is implementation on parallel hardware.

Gabor filtering is spatially-parallel and largely data-independent, and we noted above that these attributes render it suitable for implementation on SIMD² or special-purpose hardware (§4.1; §A.1). It is reasonable to assume that this approach could cause the overhead arising from Gabor filtering to be negligible. By contrast, the processing requirements of most Markov algorithms are spatially-localised and data-dependent, and hence are less straightforward to implement efficiently. Previous approaches to the problem of mapping Gibbs-Markov algorithms to SIMD or systolic hardware certainly improve upon sequential implementations, but are not entirely satisfactory (§5.6.6). The Markov property implies that well-separated image codes may be updated independently, and since the MRF neighbourhood size is small compared to the dimensions of the image lattice, this allows some parallelism to be extracted. During image segmentation, however, processing requirements are far from homogeneous across the image, and there is a danger that processor utilisation will be low. The more flexible MIMD³ paradigm is more appropriate in this instance (§A.1.2).

This provisional analysis suggests that Gabor-Markov models require support for both SIMD and MIMD processing modes, and hence are implemented most efficiently on *dual-paradigm* parallel hardware. A suitable architecture, the DisPuter, is described in §A.1.3 (Winder, 1988a). Further work is required to establish appropriate partitions of image and data between the two paradigms. Particular attention needs to be given to communication requirements, as our experience suggests that these may impose more severe constraints on performance than computation alone (Winder, 1987). Once real-time performance seems attainable, further research should address adaptation of existing static algorithms to exploit motion coherence.

²Single Instruction Multiple Data, or "data-parallel".

³Multiple Instruction Multiple Data, or "task-parallel".

Bibliography

- AHUJA, N. AND ROSENFELD, A. (1981). Mosaic models for textures. *IEEE Transactions on Pattern Analysis and Machine Intelligence* PAMI-3, 1-11.
- AMBLER, B. A. AND FINKLEA, D. L. (1976). The influence of selective attention in peripheral and foveal vision. *Perception & Psychophysics* 19, 518-524.
- ANSTIS, S. M. (1974). A chart demonstrating variations in acuity with retinal position. *Vision Research* 14, 589-592.
- BALLARD, D. H. (1984). Parameter nets. *Artificial Intelligence* 22, 235-267.
- BALLARD, D. H. AND BROWN, C. M. (1982). *Computer Vision*. Prentice Hall, New Jersey.
- BARLOW, H. B. (1986). Why have multiple cortical areas? *Vision Research* 26, 81-90.
- BARROW, H. G. AND TENENBAUM, J. M. (1981). Computational vision. *Proceedings of the IEEE* 69, 572-595.
- BATEMAN, A. AND YATES, W. (1988). *Digital Signal Processing Design*, chapter 2, pages 43-85. Pitman, London.
- BECK, J. (1983). Textural segmentation, second-order statistics, and textural elements. *Biological Cybernetics* 48, 125-130.
- BECK, J. AND AMBLER, B. A. (1972). Discriminability of differences in line slope and in line arrangements as a function of mask delay. *Perception & Psychophysics* 12, 33-38.
- BECK, J., PRAZDNY, K., AND ROSENFELD, A. (1983). A theory of textural segmentation. In Beck, J., Hope, B., and Rosenfeld, A., editors, *Human and Machine Vision*, pages 1-38. Academic Press, London.
- BECK, J., ROSENFELD, A., AND IVRY, R. (1989). Line segregation. *Spatial Vision* 4, 75-101.
- BERGEN, J. R. AND ADELSON, E. H. (1988). Early vision and texture perception. *Nature* 333, 363-364.
- BERGEN, J. R. AND JULESZ, B. (1983). Rapid discrimination of visual patterns. *IEEE Transactions on Systems, Man and Cybernetics* SMC-13, 857-863.
- BESAG, J. E. (1974). Spatial interaction and the statistical analysis of lattice systems. *Journal of the Royal Statistical Society B* 36, 192-236. With Discussion.
- BESAG, J. E. (1986). On the statistical analysis of dirty pictures. *Journal of the Royal Statistical Society B* 48, 259-302. With Discussion.
- BHANU, B. AND PARVIN, B. A. (1987). Segmentation of natural scenes. *Pattern Recognition* 20, 487-496.

- BLANZ, W. E. AND REINHARDT, E. R. (1981). Image segmentation by pixel classification. *Pattern Recognition* 13, 293-298.
- BLOSTEIN, D. AND AHUJA, N. (1987). Representation and three-dimensional interpretation of image texture: An integrated approach. In Brady, J. M. and Rosenfeld, A., editors, *First International Conference on Computer Vision*, pages 444-449, London. IEEE.
- BOUMAN, C. AND LIU, B. (1991). Multiple resolution segmentation of textured images. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 13, 99-113.
- BOVIK, A. C., CLARK, M., AND GIESLER, W. S. (1990). Multichannel texture analysis using localized spatial filters. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 12, 55-73.
- BRADY, J. M. (1981). Preface — the changing shape of computer vision. *Artificial Intelligence* 17, 1-15.
- BROADBENT, D. E. (1987). Simple models for experimentable situations. In Morris, P., editor, *Modelling Cognition*, chapter 6, pages 169-185. John Wiley.
- BRODATZ, P. (1966). *Textures: A Photographic Album for Artists and Designers*. Dover, Toronto.
- BURT, P. J. AND ADELSON, E. H. (1983). The Laplacian pyramid as compact image code. *IEEE Transactions on Communications* COM-31, 532-540.
- CAELLI, T. M. (1985). Three processing characteristics of visual texture segmentation. *Spatial Vision* 1, 19-30.
- CAELLI, T. M. (1988). An adaptive computational model for texture segmentation. *IEEE Transactions on Systems, Man and Cybernetics* 18, 9-17.
- CALLAGHAN, T. C., LASAGA, M. I., AND GARNER, W. R. (1986). Visual texture segregation based on orientation and hue. *Perception & Psychophysics* 39, 32-38.
- CANNY, J. F. (1983). Finding edges and lines in images. Technical Report AI-TR-720, MIT Artificial Intelligence Laboratory, Cambridge, Massachusetts.
- CAVE, K. R. AND WOLFE, J. M. (1990). Modeling the role of parallel processing in visual search. *Cognitive Psychology* 22, 225-271.
- CHELLAPPA, R. AND CHATTERJEE, S. (1985). Classification of textures using Gaussian Markov random fields. *IEEE Transactions on Acoustics, Speech, and Signal Processing* ASSP-33, 959-963.
- CHELLAPPA, R. AND KASHYAP, R. L. (1982). Digital image restoration using spatial interaction models. *IEEE Transactions on Acoustics, Speech, and Signal Processing* ASSP-30, 461-472.
- CHEN, P. C. AND PAVLIDIS, T. (1983). Segmentation by texture using correlation. *IEEE Transactions on Pattern Analysis and Machine Intelligence* PAMI-5, 64-69.
- CLARK, M., BOVIK, A. C., AND GIESLER, W. S. (1987). Texture segmentation using Gabor modulation/demodulation. *Pattern Recognition Letters* 8, 261-267.
- COGGINS, J. M. AND JAIN, A. K. (1985). A spatial filtering approach to texture analysis. *Pattern Recognition Letters* 3, 195-203.

- COHEN, F. S. AND COOPER, D. B. (1987). Simple parallel hierarchical and relaxation algorithms for segmenting noncausal Markovian random fields. *IEEE Transactions on Pattern Analysis and Machine Intelligence* PAMI-9, 195-219.
- COHEN, F. S., FAN, Z., AND PATEL, M. A. (1991). Classification of rotated and scaled textured images using Gaussian Markov random field models. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 13, 192-202.
- COLEMAN, G. B. AND ANDREWS, H. C. (1979). Image segmentation by clustering. *Proceedings of the IEEE* 67, 773-785.
- CONNERS, R. W. AND HARLOW, C. A. (1980). A theoretical comparison of texture algorithms. *IEEE Transactions on Pattern Analysis and Machine Intelligence* PAMI-2, 204-222.
- COWEY, A. (1985). Organization related to selective attention and selective impairments of visual perception: A tutorial review. In Posner, M. I. and Mariu, O. S. M., editors, *Attention and Performance XI*, chapter 2, pages 41-62. Lawrence Erlbaum, London.
- CRASSINI, B. (1986). On the spatial distribution of visual attention. *Journal of Experimental Psychology: Human Perception and Performance* 12, 380-382.
- CROSS, G. R. AND JAIN, A. K. (1983). Markov random field texture models. *IEEE Transactions on Pattern Analysis and Machine Intelligence* PAMI-5, 25-39.
- DAUGMAN, J. G. (1985). Uncertainty relation for resolution in space, spatial frequency, and orientation optimized by two-dimensional visual cortical filters. *Journal of the Optical Society of America A* 2, 1160-1169.
- DAVIS, L. S., CLEARMAN, M., AND AGGARWAL, J. K. (1981). An empirical evaluation of generalized cooccurrence matrices. *IEEE Transactions on Pattern Analysis and Machine Intelligence* PAMI-3, 214-221.
- DAVIS, L. S., JOHNS, S. A., AND AGGARWAL, J. K. (1979). Texture analysis using generalized co-occurrence matrices. *IEEE Transactions on Pattern Analysis and Machine Intelligence* PAMI-1, 251-259.
- DERIN, H. AND COLE, W. S. (1986). Segmentation of textured images using Gibbs random fields. *Computer Vision, Graphics and Image Processing* 35, 72-98.
- DERIN, H. AND ELLIOTT, H. (1987). Modeling and segmentation of noisy and textured images using Gibbs random fields. *IEEE Transactions on Pattern Analysis and Machine Intelligence* PAMI-9, 39-55.
- DERIN, H. AND WON, C.-S. (1987). A parallel image segmentation algorithm using relaxation with varying neighborhoods and its mapping to array processors. *Computer Vision, Graphics and Image Processing* 40, 54-78.
- DEVIJVER, P. A. AND KITTLER, J. (1982). *Pattern Recognition: A Statistical Approach*. Prentice Hall, New Jersey.
- DU BUF, J. M. H., KARDAN, M., AND SPANN, M. (1990). Texture feature performance for image segmentation. *Pattern Recognition* 23, 291-309.
- DUNCAN, J. AND HUMPHREYS, G. W. (1989). Visual search and stimulus similarity. *Psychological Review* 96, 433-458.
- ENNS, J. T. (1986). Seeing textons in context. *Perception & Psychophysics* 39, 143-147.

- EOM, K. B. AND KASHYAP, R. L. (1990). Composite edge detection with random field models. *IEEE Transactions on Systems, Man and Cybernetics* **20**, 81-93.
- ERIKSEN, C. W. AND MURPHY, T. D. (1987). Movement of attentional focus across the visual field: A critical look at the evidence. *Perception & Psychophysics* **42**, 299-305.
- ERIKSEN, C. W. AND ST. JAMES, J. D. (1986). Visual attention within and around the field of focal attention: A zoom lens model. *Perception & Psychophysics* **40**, 225-240.
- FAUGERAS, O. D. AND PRATT, W. K. (1980). Decorrelation methods of texture feature extraction. *IEEE Transactions on Pattern Analysis and Machine Intelligence PAMI-2*, 323-332.
- FOGEL, I. AND SAGI, D. (1989). Gabor filters as texture discriminator. *Biological Cybernetics* **61**, 103-113.
- GABOR, D. (1946). Theory of communication. *Journal of the IEE* **93**(III), 429-457.
- GAGALOWICZ, A. AND MA, S. D. (1985). Sequential synthesis of natural textures. *Computer Vision, Graphics and Image Processing* **30**, 289-315.
- GEMAN, S. AND GEMAN, D. (1984). Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images. *IEEE Transactions on Pattern Analysis and Machine Intelligence PAMI-6*, 721-741.
- GILMORE, G. C. (1985). Letters are visual stimuli: A comment on the use of confusion matrices. *Perception & Psychophysics* **37**, 389-390.
- GURARI, E. M. AND WECHSLER, H. (1982). On the difficulties involved in the segmentation of pictures. *IEEE Transactions on Pattern Analysis and Machine Intelligence PAMI-4*, 304-306.
- HAMEY, L. G. C. AND KANADE, T. (1989). Computer analysis of regular repetitive textures. In *Proceedings of the DARPA Image Understanding Workshop*, pages 1076-1088, Palo Alto, California. Morgan Kaufmann.
- HANSEN, F. R. AND ELLIOTT, H. (1982). Image segmentation using simple Markov field models. *Computer Graphics and Image Processing* **20**, 101-132.
- HARALICK, R. M. (1979). Statistical and structural approaches to texture. *Proceedings of the IEEE* **67**, 786-804.
- HARALICK, R. M., SHANMUGAM, K., AND DINSTEN, I. (1973). Textural features for image classification. *IEEE Transactions on Systems, Man and Cybernetics SMC-3*, 610-621.
- HASSNER, M. AND SKLANSKY, J. (1980). The use of Markov random fields as models of texture. *Computer Graphics and Image Processing* **12**, 357-370.
- HEEGER, D. J. (1987). Model for the extraction of image flow. *Journal of the Optical Society of America A* **4**, 1455-1471.
- HOFFMAN, J. E. (1979). A two-stage model of visual search. *Perception & Psychophysics* **25**, 319-327.
- HSIAO, J. Y. AND SAWCHUK, A. A. (1989). Supervised textured image segmentation using feature smoothing and probabilistic relaxation techniques. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **11**, 1279-1292.
- HUET, B. R. AND CANNON, T. M. (1976). Nonstationary assumptions for Gaussian models of images. *IEEE Transactions on Systems, Man and Cybernetics SMC-6*, 876-882.

- JÁÑEZ, L. (1984). Visual grouping without low spatial frequencies. *Vision Research* **24**, 271-274.
- JULESZ, B. (1962). Visual pattern discrimination. *IRE Transactions on Information Theory* **IT-8**, 84-92.
- JULESZ, B. (1975). Experiments in the visual perception of texture. *Scientific American* **232**(4), 34-43.
- JULESZ, B. (1980). Spatial nonlinearities in the instantaneous perception of textures with identical power spectra. *Philosophical Transactions of the Royal Society of London B* **290**, 83-94.
- JULESZ, B. (1981). Textons, the elements of texture perception, and their interactions. *Nature* **290**, 91-97.
- JULESZ, B. (1984). A brief outline of the texton theory of human vision. *Trends in Neurosciences* **7**, 41-45.
- JULESZ, B. (1986). Texton gradients: The texton theory revisited. *Biological Cybernetics* **54**, 245-251.
- JULESZ, B. AND BERGEN, J. R. (1983). Textons, the fundamental elements in prattentive vision and perception of textures. *Bell Systems Technical Journal* **62**, 1619-1645.
- JULESZ, B., GILBERT, E. N., AND VICTOR, J. D. (1978). Visual discrimination of textures with identical third-order statistics. *Biological Cybernetics* **31**, 137-140.
- JULESZ, B. AND KRÖSE, B. J. A. (1988). Features and spatial filters. *Nature* **333**, 302-303.
- KASHYAP, R. L. AND CHELLAPPA, R. (1983). Estimation and choice of neighbors in spatial-interaction models of images. *IEEE Transactions on Information Theory* **IT-29**, 60-72.
- KASHYAP, R. L., CHELLAPPA, R., AND KHOTANZAD, A. (1982). Texture classification using features derived from random field models. *Pattern Recognition Letters* **1**, 43-50.
- KASHYAP, R. L. AND KHOTANZAD, A. (1986). A model-based method for rotation-invariant texture classification. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **PAMI-8**, 472-481.
- KASS, M. AND WITKIN, A. P. (1985). Analyzing oriented patterns. In *Ninth International Joint Conference on Artificial Intelligence*, pages 944-952, Los Angeles.
- KHOTANZAD, A. AND CHEN, J.-Y. (1989). Unsupervised segmentation of textured images by edge detection in multidimensional features. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **11**, 414-421.
- KHOTANZAD, A. AND KASHYAP, R. L. (1987). Feature selection for texture recognition based on image synthesis. *IEEE Transactions on Systems, Man and Cybernetics* **SMC-17**, 1087-1095.
- KIRKPATRICK, S., GELATT, C. D., AND VECCHI, M. P. (1983). Optimization by simulated annealing. *Science* **220**, 671-680.
- KITTLER, J. AND ILLINGWORTH, J. (1985). Relaxation labelling algorithms — a review. *Image and Vision Computing* **3**, 206-216.
- KJELL, B. P. AND DYER, C. R. (1985). Edge separation and orientation texture measures. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 306-311, San Francisco. IEEE.

- LAKSHMANAN, S. AND DERIN, H. (1989). Simultaneous parameter estimation and segmentation of Gibbs random fields using simulated annealing. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 11, 799-813.
- LIVINGSTONE, M. S. AND HUBEL, D. H. (1987). Psychophysical evidence for separate channels for the perception of form, color, movement, and depth. *Journal of Neuroscience* 7, 3416-3468.
- LOWE, D. G. (1987). Three-dimensional object recognition from single two-dimensional images. *Artificial Intelligence* 31, 355-395.
- MALIK, J. AND PERONA, P. (1990). Preattentive texture discrimination with early vision mechanisms. *Journal of the Optical Society of America A* 7, 923-932.
- MALLAT, S. G. (1989). Multifrequency channel decompositions of images and wavelet models. *IEEE Transactions on Acoustics, Speech, and Signal Processing* 37, 2091-2110.
- MANJUNATH, B. S. AND CHELLAPPA, R. (1991). Unsupervised texture segmentation using Markov random field models. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 13, 478-482.
- MARR, D. C. (1976). Early processing of visual information. *Philosophical Transactions of the Royal Society of London B* 275, 483-519.
- MARR, D. C. (1982). *Vision: A Computational Investigation into the Human Representation and Processing of Visual Information*. W.H. Freeman, San Francisco.
- MARR, D. C. AND HILDRETH, E. C. (1980). Theory of edge detection. *Proceedings of the Royal Society of London B* 207, 187-217.
- MEER, P., BAUGHER, E. S., AND ROSENFELD, A. (1987). Frequency domain analysis and synthesis of image pyramid generating kernels. *IEEE Transactions on Pattern Analysis and Machine Intelligence* PAMI-9, 512-522.
- MITCHELL, O. R., MYERS, C. R., AND BOYNE, W. (1977). A max-min measure for image texture analysis. *IEEE Transactions on Computers* C-26, 408-414.
- MODESTINO, J. W., FRIES, R. W., AND VICKERS, A. L. (1981). Texture discrimination based upon an assumed stochastic texture model. *IEEE Transactions on Pattern Analysis and Machine Intelligence* PAMI-3, 557-580.
- MORAN, J. AND DESIMONE, R. (1985). Selective attention gates visual processing in the extrastriate cortex. *Science* 229, 782-784.
- MULLER, M. J. (1986). Texture boundaries: Important cues for human texture discrimination. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 464-468, Miami Beach. IEEE.
- MUMFORD, D., KOSSLYN, S. M., HILLGER, L. A., AND HERRNSTEIN, R. J. (1987). Discriminating figure from ground: The role of edge detection and region growing. *Proceedings of the National Academy of Sciences of the USA* 84, 7354-7358.
- MURRAY, D. W., KASHKO, A., AND BUXTON, H. (1986). A parallel approach to the picture restoration algorithm of Geman and Geman on an SIMD machine. *Image and Vision Computing* 4, 133-142.
- NAHI, N. E. AND JARANSHARI, M. H. (1977). Image boundary estimation. *IEEE Transactions on Computers* C-26, 772-781.

- NISSEN, M. J. (1985). Accessing features and objects: Is location special? In Posner, M. I. and Marin, O. S. M., editors, *Attention and Performance XI*, chapter 10, pages 205-219. Lawrence Erlbaum, London.
- NOTHDURFT, H. C. (1990). Texton segregation by associated differences in global and local luminance distribution. *Proceedings of the Royal Society of London B* **230**, 295-320.
- PAPOULIS, A. (1984). *Probability, Random Variables, and Stochastic Processes*. McGraw Hill, New York, second edition.
- PAPOULIS, A. (1990). *Probability and Statistics*. Prentice Hall, London.
- PAPOULIS, A. (1991). *Probability, Random Variables, and Stochastic Processes*. McGraw Hill, New York, third edition.
- PASHLER, H. (1988). Cross-dimensional interaction and texture segregation. *Perception & Psychophysics* **43**, 307-318.
- PELEG, S., NAOR, J., HARTLEY, R., AND AVNIR, D. (1984). Multiple resolution texture analysis and classification. *IEEE Transactions on Pattern Analysis and Machine Intelligence PAMI-6*, 518-523.
- PERRY, A. AND LOWE, D. G. (1989). Segmentation of textured images. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 319-325, San Diego. IEEE.
- PIETIKÄINEN, M., ROSENFELD, A., AND DAVIS, L. S. (1983). Experiments with texture classification using averages of local pattern matches. *IEEE Transactions on Systems, Man and Cybernetics SMC-13*, 421-426.
- POLLEN, D. A. AND RONNER, S. F. (1981). Phase relationships between adjacent simple cells in the visual cortex. *Science* **212**, 1409-1411.
- PORAT, M. AND ZEEVI, Y. Y. (1988). The generalised Gabor scheme of image representation in biological and machine vision. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **10**, 452-468.
- POSNER, M. I. (1980). Orienting of attention. *Quarterly Journal of Experimental Psychology* **32**, 3-25.
- POSNER, M. I. AND PRESTI, D. E. (1987). Selective attention and cognitive control. *Trends in Neurosciences* **10**, 13-17.
- POSNER, M. I., SNYDER, C. R. R., AND DAVIDSON, B. J. (1980). Attention and the detection of signals. *Journal of Experimental Psychology: General* **109**, 160-174.
- PRATT, W. K., FAUGERAS, O. D., AND GAGALOWICZ, A. (1978). Visual discrimination of stochastic texture fields. *IEEE Transactions on Systems, Man and Cybernetics SMC-8*, 796-804.
- REED, T. R. AND WECHSLER, H. (1990). Segmentation of textured images and Gestalt organization using spatial/spatial-frequency representations. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **12**, 1-12.
- REEVES, A. P. (1984). Parallel computer architectures for image processing. *Computer Vision, Graphics and Image Processing* **25**, 68-88.
- ROSENFELD, A. AND KAK, A. C. (1982). *Digital Picture Processing*. Academic Press, London, second edition.

- RUBENSTEIN, B. S. AND SAGI, D. (1990). Spatial variability as a limiting factor in texture-discrimination tasks: implications for performance asymmetries. *Journal of the Optical Society of America A* 7, 1632-1643.
- SAGI, D. AND JULESZ, B. (1984). Detection versus discrimination of visual orientation. *Perception* 13, 619-628.
- SAGI, D. AND JULESZ, B. (1985). "Where" and "what" in vision. *Science* 228, 1217-1219.
- SHER, C. A. AND ROSENFELD, A. (1989). Detecting and extracting compact textured regions using pyramids. *Image and Vision Computing* 7, 129-134.
- SILVERMAN, J. F. AND COOPER, D. B. (1988). Bayesian clustering for unsupervised estimation of surface and texture models. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 10, 482-495.
- SPACEK, L. A. (1986). Edge detection and motion detection. *Image and Vision Computing* 4, 43-56.
- SPANN, M. AND WILSON, R. (1985). A quad-tree approach to image segmentation which combines statistical and spatial information. *Pattern Recognition* 18, 257-269.
- SPITZER, H., DESIMONE, R., AND MORAN, J. (1988). Increased attention enhances both behavioral and neuronal performance. *Science* 240, 338-340.
- STULLER, J. A. AND KURZ, B. (1976). Two-dimensional Markov representations of sampled images. *IEEE Transactions on Communications COM-24*, 1148-1152.
- SUTTER, A. K., BECK, J., AND GRABAM, N. (1989). Contrast and spatial variables in texture segregation: Testing a simple spatial-frequency channels model. *Perception & Psychophysics* 46, 312-332.
- TAMURA, H., MORI, S., AND YAMAWAKI, T. (1978). Textural features corresponding to visual perception. *IEEE Transactions on Systems, Man and Cybernetics SMC-8*, 460-473.
- TAN, T. (1988). Texture edge detection via Gabor filtering. Technical Report TR-08-TNT, Department of Electrical Engineering, Imperial College, London.
- TERZOPOULOS, D. (1986). Image analysis using multigrid relaxation methods. *IEEE Transactions on Pattern Analysis and Machine Intelligence PAMI-8*, 129-139.
- TOMITA, F., SHIRAI, Y., AND TSUJI, S. (1982). Description of textures by a structural analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence PAMI-4*, 183-191.
- TREISMAN, A. M. (1982). Perceptual grouping and attention in visual search for features and for objects. *Journal of Experimental Psychology: Human Perception and Performance* 8, 194-214.
- TREISMAN, A. M. (1985). Preattentive processing in vision. *Computer Vision, Graphics and Image Processing* 31, 156-177.
- TREISMAN, A. M. AND GELADE, G. (1980). A feature-integration theory of attention. *Cognitive Psychology* 12, 97-136.
- TREISMAN, A. M. AND GORMICAN, S. (1988). Feature analysis in early vision: Evidence from search asymmetries. *Psychological Review* 95, 15-48.

- TREISMAN, A. M. AND PATERSON, R. (1984). Emergent features, attention, and object perception. *Journal of Experimental Psychology: Human Perception and Performance* 10, 12-31.
- TREISMAN, A. M. AND SATO, S. (1990). Conjunction search revisited. *Journal of Experimental Psychology: Human Perception and Performance* 16, 459-478.
- TREISMAN, A. M. AND SOUTHER, J. (1985). Search asymmetry: A diagnostic for preattentive processing of separable features. *Journal of Experimental Psychology: General* 114, 285-310.
- TRUSSELL, H. J. AND KRUGER, R. P. (1978). Comments on "Nonstationary assumptions for Gaussian models in images". *IEEE Transactions on Systems, Man and Cybernetics* SMC-8, 579-582.
- TSOTSOS, J. K. (1987). A 'complexity level' analysis of vision. In Brady, J. M. and Rosenfeld, A., editors, *First International Conference on Computer Vision*, pages 346-355, London. IEEE.
- TURNER, M. R. (1986). Texture discrimination by Gabor functions. *Biological Cybernetics* 55, 71-82.
- ULLMAN, S. (1984). Visual routines. *Cognition* 18, 97-159. Re-printed from: Massachusetts Institute of Technology Artificial Intelligence Memo 723.
- UNSER, M. (1986). Sum and difference histograms for texture classification. *IEEE Transactions on Pattern Analysis and Machine Intelligence* PAMI-8, 118-125.
- UNSER, M. AND EDEN, M. (1989). Multiresolution feature extraction and selection for texture segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 11, 717-728.
- VICKERS, A. L. AND MODESTINO, J. W. (1982). A maximum likelihood approach to texture classification. *IEEE Transactions on Pattern Analysis and Machine Intelligence* PAMI-4, 61-68.
- VILNROTTER, F. M., NEVATIA, R., AND PRICE, K. E. (1986). Structural analysis of natural textures. *IEEE Transactions on Pattern Analysis and Machine Intelligence* PAMI-8, 76-89.
- VOORHEES, H. L. (1987). Finding texture boundaries in images. Master's thesis, Massachusetts Institute of Technology. Re-printed as: Massachusetts Institute of Technology Technical Report 968.
- VOORHEES, H. L. AND POGGIO, T. (1987). Detecting textures and texture boundaries in natural images. In Brady, J. M. and Rosenfeld, A., editors, *First International Conference on Computer Vision*, pages 250-258, London. IEEE.
- WATT, R. J. (1988). *Visual Processing: Computational, Psychophysical and Cognitive Research*. Lawrence Erlbaum, London.
- WESZKA, J. S., DYER, C. R., AND ROSENFELD, A. (1976). A comparative study of texture measures for terrain classification. *IEEE Transactions on Systems, Man and Cybernetics* SMC-6, 269-285.
- WILSON, H. R. (1983). Psychophysical evidence for spatial channels. In Braddick, O. J. and Sleigh, A. C., editors, *Physical and Biological Processing of Images*, pages 88-99. Springer-Verlag, Berlin.

- WILSON, R. AND GRANLUND, G. H. (1984). The uncertainty principle in image processing. *IEEE Transactions on Pattern Analysis and Machine Intelligence PAMI-6*, 758-767.
- WINDER, C. P. (1986). Transputer upgrade of the Disarray graphics processor. M.Sc. dissertation, Programming Research Group, Oxford University Computing Laboratory, Oxford.
- WINDER, C. P. (1987). Bouncing balls on the Disputer. Technical Report PRG-TR-2-87,¹ Programming Research Group, Oxford University Computing Laboratory, Oxford.
- WINDER, C. P. (1988a). Parallel processing with the Disputer. In Kerridge, J., editor, *Developments Using occam*, pages 31-43. I. O. S., Amsterdam. Presented at the 8th Technical Meeting of the occam User Group, Sheffield, March 1988.
- WINDER, C. P. (1988b). Visual perceptual attention: Towards a computational model for feature-integration theory. Technical Report PRG-TR-1-88,¹ Programming Research Group, Oxford University Computing Laboratory, Oxford.
- WINDER, C. P. (1989). Where's the psychological reality? *Behavioral and Brain Sciences* **12**, 417.
- WINDER, C. P. (1990). Spatial scale and perceptual grouping in human visual attention. Technical Report PRG-TR-1-90,¹ Programming Research Group, Oxford University Computing Laboratory, Oxford.
- WOLFE, J. M. AND CAVE, K. R. (1990). Deploying visual attention: The guided search model. In Blake, A. and Troscianko, T., editors, *AI and the Eye*, pages 79-103. John Wiley, Chichester.
- WOLFE, J. M., CAVE, K. R., AND FRANZEL, S. L. (1989). Guided search: An alternative to the feature integration model for visual search. *Journal of Experimental Psychology: Human Perception and Performance* **15**, 419-433.
- WOLFE, J. M., YU, K. P., STEWART, M. I., SHORTER, A. D., FRIEDMAN-HILL, S. R., AND CAVE, K. R. (1990). Limitations on the parallel guidance of visual search: Color \times color and orientation \times orientation conjunctions. *Journal of Experimental Psychology: Human Perception and Performance* **16**, 879-892.
- WOODS, J. W. (1972). Two-dimensional discrete Markovian fields. *IEEE Transactions on Information Theory IT-18*, 232-240.
- ZAKHAROV, V. (1984). Parallelism and array processing. *IEEE Transactions on Computers C-33*, 45-78.
- ZEKI, S. M. AND SHIPP, S. (1988). The functional logic of cortical connections. *Nature* **335**, 311-317.
- ZUCKER, S. W. (1983). Computational and psychophysical experiments in grouping: Early orientation selection. In Beck, J., Hope, B., and Rosenfeld, A., editors, *Human and Machine Vision*, pages 545-567. Academic Press, London.
- ZUCKER, S. W., ROSENFELD, A., AND DAVIS, L. S. (1975). Picture segmentation by texture discrimination. *IEEE Transactions on Computers C-24*, 1228-1233.

¹Copies available on request from: The Librarian, Oxford University Computing Laboratory, 8-11 Keble Road, Oxford OX1 3QD, United Kingdom.

Glossary of Principal Notation

Our principal notation is given below, but additional local terminology is defined in the main text.

Abbreviations

1-D	one dimensional
2-D	two dimensional
CPU	Central Processing Unit
DC	Direct Current; implying “spatially-constant” in this context
DOOG	Difference Of Oriented Gaussians
FFT	Fast Fourier Transform
GMRF	Gibbs–Markov Random Field
GRF	Gibbs Random Field
LHS	Left Hand Side
LINE	Gabor image primitive (§4.5)
MAP	Maximum A Posteriori
MIMD	Multiple Instruction Multiple Data
MRF	Markov Random Field
PE	Processor Element
RHS	Right Hand Side
RT	Reaction Time
SAR	Simultaneous AutoRegressive (random field models)
SIMD	Single Instruction Multiple Data
SINU	Gabor image primitive (§4.5)
VLSI	Very Large Scale Integration

Other Terminology

#55	plate number of a Brodatz texture (Figure 1.6 on page 11)
§	section
All-Quarters	texture classification paradigm (§6.4)
Group	texture classification paradigm (§6.4)
Profile	Gabor feature-extraction method (§4.6)
Resultant	Gabor feature-extraction method (§4.6)
SEG, POP	artificial “segregation” and “pop-out” displays (Figure 4.5 on page 71)

Universal Notation

x, X	random numbers (Euler font)
\mathbf{X}, Σ	matrices (uppercase bold sans-serif), with elements $X_{k,l}$
\mathcal{X}, \mathcal{S}	sets (uppercase script)
\mathbf{x}, \mathbf{X}	vectors (bold), with elements x_i
\mathbf{x}, \mathbf{X}	random vectors (bold Euler)
x, X, α	scalar variables (math italic)
$\mathcal{P}\{A\}$	probability of the event A
$\mathcal{P}\{A B\}$	probability of the event A conditional on event B
$\mathcal{E}\{x\}$	mathematical expectation of the random variable x
$x * y$	discrete linear convolution of functions x and y
A^*	complex conjugate of A
\mathbf{x}^T	transpose of vector \mathbf{x}
\mathbf{x}^\dagger	conjugate transpose of vector \mathbf{x}
$\#A, \#A$	cardinality of the vector \mathbf{A} or set A
$\ \mathbf{A}\ $	Euclidean norm of the vector \mathbf{A} : $\ \mathbf{A}\ ^2 = \sum_i A_i ^2$
$\det \mathbf{A}, \mathbf{A} $	determinant of matrix \mathbf{A}
$x = X$	denotes that functions x and X are Fourier transform pairs
$a \sim b$	denotes that a and b are similarly distributed
$\mathcal{F}(x) = X$	Fourier transform of function x
\bar{A}	sample mean of variable A
$\text{var } x$	variance of variable x
$\mathbf{N}, \mathbf{R}, \mathbf{C}$	sets of natural, real and complex numbers
j	positive root of -1
\mathcal{F}	Fourier matrix
$\mathbf{1}$	the identity matrix (rank determined by context)
$\mathbf{0}$	constant vector, all elements 0 (rank determined by context)
$\mathbf{1}$	constant vector, all elements 1 (rank determined by context)
$\delta(\cdot), \delta_{k,l}$	Dirac delta-function
$\Phi(\cdot)$	inverse Normal distribution
$\mathcal{G}(\mu, \sigma^2)$	normal distribution with mean μ and variance σ^2
$\mathcal{O}(\cdot)$	computational complexity of algorithm
\mathcal{L}	image lattice
s_x^2	sample variance of variable x
$I(x, y)$	two-dimensional image array
$g(\zeta) = G(\zeta)$	Gaussian distributions, space constant ζ
\mathcal{V}	set of allowable gray levels
θ, c, ϵ	Gabor orientation, contrast and error features
J_B	Bhattacharyya coefficient
Δ_M	Mahalanobis distance
Δ_W	weighted Euclidean distance
\mathcal{S}	state space: set of feasible lattice configurations
\mathbf{R}_θ	rotation matrix: $\mathbf{R}_\theta = \begin{pmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{pmatrix}$

Gabor Notation

The following notation is used primarily in Chapter 4, and also in Appendices A.2 and A.3.

$\Delta t, \Delta f$	measurement uncertainty for time t and temporal frequency f
$\mathbf{x} = (x, y)^T$	two-dimensional axes
$\mathbf{x}' = (x', y')^T$	rotated two-dimensional axes
$\mathbf{u} = (u, v)^T$	two-dimensional spatial-frequency axes
$q = Q$	complex Gabor kernel, parameters given below
q_c, q_s	cosine and sine phases of Gabor kernel: $q = q_c + jq_s$
R	model complex Gabor response
\mathbf{R}	model complex Gabor response from several channels
M, \mathbf{M}	measured complex Gabor response
R_c, R_s, M_c, M_s	phase components of Gabor responses: $R = R_c + jR_s$
$\mathcal{R}, \mathcal{M}, \mathcal{R}, \mathcal{M}$	Gabor energy responses: $\mathcal{R} = \overline{R_c^2 + R_s^2}$
A	Gabor amplitude: $A = \sqrt{R_c^2 + R_s^2}$
ϕ, Φ	model Gabor phase response: $\phi = \arctan(R_s/R_c)$
P, \mathbf{P}	measured Gabor phase response
$i_{k,t}$	coefficients of Gabor image representation
$f_{\text{LINE}}, f_{\text{SINU}}$	definitions for LINE and SINU primitives
d, c, θ_0	offset, contrast, and orientation parameters for LINE primitive
$\varpi, c, \psi, \theta_0$	angular frequency, contrast, phase, and orientation parameters for SINU primitive
ϑ	relative orientation between filter and image primitive: $\vartheta = \theta_0 - \theta$
$k_{\text{DC}}(\sigma_{\text{DC}})$	image pre-processing filter
ε	signature match error
ξ, ξ_{max}	relative sampling error
ρ, ρ_{max}	sampling ratio (relative to Nyquist frequency)
α	$\sqrt{\ln 4}$
T_s	sampling period for Gabor image representations

Filter Parameters

$\Delta\omega_{1/2}$	half-height angular-frequency bandwidth
$B_{1/2}$	octave half-height spatial-frequency bandwidth
$\Delta\theta_{1/2}$	half-height orientation bandwidth
T	sinusoid period
ω	sinusoid angular frequency
λ	envelope aspect ratio
θ	filter orientation ($0 \leq \theta < \pi$)
σ	envelope space constant
Θ	angular separation of Gabor filters
ς	smoothing parameter (Gaussian space-constant)

Gibbs–Markov Notation

This notation is used primarily in Chapters 5–7, and also in Appendix A.5.

R_{xx}, R_{xx}	autocorrelation of process x
C_{xx}, C_{xx}	autocovariance of process x
Γ_{xx}, Γ_{xx}	Fourier transform of autocorrelation: $\Gamma_{xx} \rightleftharpoons R_{xx}$
S_{xx}	spectral density of stationary process x : $S_{xx}(\omega) \rightleftharpoons R_{xx}(\tau)$
$F(x)$	distribution function for x : $F(x) = \mathcal{P}\{x \leq x\}$
$f(x)$	density function for x : $f(x) = \partial F(x)/\partial x$
$\mathcal{N}_{\mathcal{P}}$	non-causal Markov neighbourhood of order \mathcal{P}
$\mathcal{N}^+, \mathcal{N}^c$	unilateral and causal Markov neighbourhoods
S	span of unilateral Markov neighbourhood \mathcal{N}^+
Ω_n	set of all image sites except n : $\Omega_n = \mathcal{L} \setminus \{n\}$
η_n	field neighbourhood vector for site n
$\mathcal{G}, \mathfrak{X}$	Gibbs–Markov parameter sets
\hat{x}	Markov predictor: estimate of pixel conditional on its surround
u, u	Markov predictor-error: $u = x - \hat{x}$
μ, μ	field mean
σ^2, σ_u^2	Markov predictor-error variance
β	Markov neighbourhood coefficient vector
B	cosine transform of β
B	Toeplitz coefficient matrix derived from β
$P(x), L(x)$	joint likelihood and log-likelihood for field x
p_i	probability density of site i conditional upon its surround
\tilde{P}, \tilde{L}	pseudo-likelihood and pseudo log-likelihood
$\hat{\beta}, \hat{\sigma}^2, \hat{\mu}$	true-likelihood parameter estimates
$\tilde{\beta}, \tilde{\sigma}^2$	pseudo-likelihood parameter estimates
s	sub-sampling period
f	smooth-sampling frequency
ω	angular spatial frequency
Λ, λ_i	spectral density coefficients: $\lambda_i = 1 - B_i$
$M \times N$	image dimensions
$U(x), V(x)$	Gibbs energy and potential functions
$\mathcal{G}(x)$	Gibbs distribution
b	blur parameter
ρ^2	noise variance
Z	partition function, or normalising constant
v, v	white noise
\mathcal{C}	set of all field cliques
\mathcal{Q}	an image code: $\mathcal{Q} \cap \mathcal{N}_i = \emptyset, \forall i \in \mathcal{Q} \subseteq \mathcal{L}$
\dot{x}	a centered process: zero expected mean

Gabor–Markov Notation

This notation is used primarily in Chapters 6 and 7, and also in Appendix A.4.

p, q	components of orthogonal-normal model
x, y	components of composite-feature model
τ	tangent direct-normal model



Appendices

A.1 Parallel Vision Architectures

Computers were originally introduced to replace error-prone human beings in making tedious calculations (Zakharov, 1984). Speed soon dominated reliability as the major motivation, and huge advances in technology have been made in attempts to satisfy this increasing demand but fundamental limits on the performance of single-processor architectures will soon be approached. The appeal of present single-processor supercomputers is further diminished because cost increases as an accelerating function of performance.

In this Appendix, we discuss the characteristics of *parallel* architectures, which appear to release these constraints. Parallel machines are subject to far less severe fundamental limits than individual processors, and offer a potentially linear price-performance curve. Efficient designs should satisfy performance criteria at minimum cost, suggesting that parallel architectures should be employed for large problems. Concepts of parallelism and interacting concurrent processes may help to shape software design, leading to more elegant and abstract algorithms. General-purpose multi-processor machines fall into two groups, particularly suitable for supporting data and task parallelism, respectively. Pipelined and special-purpose architectures may often be designed to meet specific requirements, but are not general-purpose and fall outside the scope of this brief review.

A.1.1 Data Parallelism: SIMD Arrays

Data-parallel machines perform the same basic operation concurrently in many processor elements (PEs), each accessing its local data. Each PE has only very limited capabilities, often operating on a single bit of data, but may be replicated many times (tens of thousands) to provide high overall processor bandwidth. Regular arrangements of simple components are particularly suitable for implementation in VLSI, permitting further reductions in cost. We shall consider SIMD¹ processor arrays, which are usually organised as a toroidal square array of PEs with local communications (“mesh-connected”). Each PE has a local memory and register set, but all instruction and data sequencing is performed by an external control unit, which broadcasts microinstruction and memory-access signals. This arrangement is very economical because unnecessary duplication of function is avoided. Its disadvantage is that each PE has very little autonomy, and all must execute the same microinstruction on the same operand (perhaps modified by local registers) at each time step: rigid translational symmetry is enforced. Numerical calculations are often performed using bit-serial arithmetic on single-bit PEs, but an increase in the *number* rather than depth of the PEs is considered to be more flexible when the extent of data parallelism exceeds the dimensions of the processor array (Reeves, 1984). Local synchronous communication achieves very high bandwidth via the local network, and random communication between PEs may be accomplished by explicit routing (very slow) or by an external routing network.

Synchrony of operation unlocks the potential of SIMD machines but imposes severe limitations on the classes of algorithm which may be executed efficiently. Early vision tasks often require large amounts of local communication but may be executed efficiently by SIMD architectures because they possess translational symmetry; convolution and relaxation are obvious examples. In the context of our discussion of low-level human vision, the SIMD paradigm is analogous to inflexible “pre-attentive” processing (Chapter 2).

Processor utilisation is a measure of the proportion of broadcast instructions that achieve useful processing, and approaches the ideal of 100% only for very well-conditioned tasks. Lower ratios are still acceptable, however, because a SIMD array is relatively cheap to implement and hence may still achieve a good price-performance ratio. When there is a strong requirement for data-specific processing, processor utilisation drops dramatically. Data-dependent processing can only be imposed locally because private PE memory is not globally accessible, and the controller must sequentially broadcast an instruction stream for all possible branches. Each PE idles until the stream appropriate to its own circumstances is received. A binary decision tree of depth N operates in $\mathcal{O}(\log N)$ time on an array of independent processors, but achieves only $\mathcal{O}(N)$ on a SIMD processor array. High-level tasks such as recognition frequently involve data-dependent processing, and hence operate with low processor utilisation. Global communication may also become a bottleneck because it is unbuffered; if one PE must receive data from all the others, this must usually be done sequentially and can be extremely slow.

Current SIMD processor arrays include the DAP, CLIP and Connection Machine. Despite the reservations given above, these are sold as general-purpose processors, and

¹Single Instruction Multiple Data.

are supported by libraries which exploit the best features of each architecture (Zakharov, 1984).

A.1.2 Task Parallelism: MIMD Networks

Processors are loosely-coupled and operate largely autonomously within MIMD² networks, which exhibit *task* parallelism. Each processor has its own program and sequencing logic, increasing its cost compared with a SIMD PE, but data-dependent branching causes no particular difficulty and MIMD networks offer much higher flexibility. There is no requirement that all processors should be identical. Communication bandwidths may be lower than for SIMD processors because of the additional requirement for synchronisation, but this comment does not apply to shared-memory architectures. Unbuffered global communication may be a bottleneck.

Although MIMD networks are more expensive to implement than SIMD arrays, they are preferred for high-level or “attentive” vision tasks, provided these do not require extensive global communication, because the greater flexibility of this paradigm permits increased efficiency. Data-parallel algorithms may also be executed conveniently on a MIMD network by replicating the same program at each processor, although a SIMD processor array could be substituted at lower cost (Reeves, 1984). A MIMD network is most appropriate when the algorithm may be broken into self-contained semi-independent tasks, and a processor farm may then be employed to achieve automatic load-balancing (Winder, 1987).

MIMD networks may easily be constructed from readily-available components, using shared memory or a global bus to support communication, and additional processors may often be added with only minor adjustment to the system. Concurrency is supported specifically by the Inmos transputer, with the intention that many transputers will be connected in a multi-processor array. Transputer point-to-point communication links are slow compared to image-processing bandwidths, suggesting that communication in transputer arrays should also be supported by a high-bandwidth global bus.

A.1.3 Dual Paradigm Parallelism: The Disputer

Data parallelism is supported efficiently by SIMD processor arrays and task parallelism by MIMD networks, but unfortunately most vision systems need to operate in *both* modes (Reeves, 1984). Neither true SIMD nor true MIMD architectures offer optimal performance because both must operate inefficiently for part of the time. The requirement for dual-paradigm parallelism is illustrated by the attentive-pre-attentive dichotomy in low-level human vision (Chapter 2).

One solution to this problem is to have a loosely-coupled array of SIMD processor arrays, and this philosophy is adopted by PASM: SIMD PEs may be swapped dynamically between modules, each supported by its own controller. There is a danger that this approach combines the expense of MIMD with the inflexibility of SIMD (Reeves, 1984).

²Multiple Instruction Multiple Data.

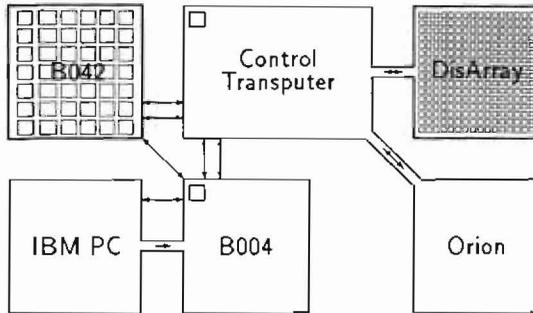


FIGURE A.1: **DisPutter Architecture.** Dual-paradigm parallelism is provided by close coupling between a 16×16 SIMD processor array *DisArray* and a transputer-based controller linked to an Inmos B042 board. Interfaces to an Orion minicomputer and an IBM PC are also supported.

Multiple SIMD is emulated by DADO: each PE may act as a conventional MIMD processor or accept instructions from a remote controller in SIMD mode, but overall efficiency is no higher than MIMD.

The DisPutter

Dual-paradigm parallel architectures provide both SIMD and MIMD components within one machine, allowing all stages of the visual pipeline to be executed efficiently and cost-effectively. We mated a 16×16 SIMD processor array *DisArray* with a transputer network to form a powerful 300-processor dual-paradigm parallel processor, the *DisPutter* (Winder, 1986). All 256 *DisArray* PEs are identical and have access to a 256 kbit local memory, local status registers and a 4-connected communications network (Figure A.1). This work began as an M.Sc. project, and the *DisPutter* was built to investigate the impact of direct support for dual-paradigm parallelism following the MIMD/SIMD model (Reeves, 1984).

DisArray microinstructions are executed globally, and take the form of an arbitrary logic function of local memory and registers. We designed and built a transputer-based microinstruction controller, achieving close coupling between SIMD and MIMD components. Additional transputers are attached by Inmos links, and other modules are accessed via EtherNet connections. Libraries were written in occam to provide low-level support for graphics and image-processing (Winder, 1987). Communication bandwidth is as important as processing bandwidth, and communication bottlenecks between the SIMD array and its transputer controller, and between transputers, are weaknesses of the *DisPutter* architecture (Winder, 1987). We also noted some undesirable features of the transputer instruction set, and proposed a small modification (Winder, 1988a).

A.2 Properties of Gabor Filters

We invoked several theoretical results concerning Gabor filters in Chapter 4 without detailed justification. The purpose of this Appendix is to provide more background to these derivations, and on the practical arrangements for Gabor filtering, and to derive Gabor signatures of simple image primitives. We adopt the notation of Chapter 4.

A.2.1 Joint Uncertainty Relation

Examination of images of physical objects requires simultaneous analysis of spatial and spectral properties. In §4.1, we claimed that Gabor filters enjoy unique advantages for this purpose, minimising the joint uncertainty with which these quantities may be measured by linear filters. Following §4.1, we shall take our Gabor filter q and its Fourier transform Q as:

$$\begin{aligned} q(\lambda, \theta, x, y) &= \frac{\lambda}{2\pi\sigma^2} \exp\left(-\frac{x'^2}{2\sigma^2}\right) \exp\left(-\frac{y'^2\lambda^2}{2\sigma^2}\right) \exp(j\omega x') \\ Q(\lambda, \theta, u, v) &= \exp\left[-\frac{1}{2}(2\pi u' - \omega)^2 \sigma^2\right] \exp\left[-\frac{1}{2}(2\pi v' \sigma / \lambda)^2\right] \end{aligned} \quad (\text{A.1})$$

where u and v represent spatial frequency along the x and y axes, respectively. The conventional definition of the uncertainty is given by the variance of the energy distribution (Daugman, 1985; Gabor, 1946), by analogy with the uncertainty principle (Wilson & Granlund, 1984):

$$\begin{aligned} (\Delta x)^2 &= \frac{\int_{-\infty}^{\infty} \int_{-\infty}^{\infty} x^2 q(x, y) q^*(x, y) dx dy}{\int_{-\infty}^{\infty} \int_{-\infty}^{\infty} q(x, y) q^*(x, y) dx dy} \\ &= \frac{\int_{-\infty}^{\infty} x^2 \exp(-x^2/\sigma^2) dx}{\int_{-\infty}^{\infty} \exp(-x^2/\sigma^2) dx} \\ &= \sigma^2/2 \\ \Delta x &= \sigma/\sqrt{2}. \end{aligned}$$

Similarly,

$$\Delta y = \frac{\sigma}{\lambda\sqrt{2}}.$$

We may follow the same procedure with $Q(u, v)$ to get:

$$\begin{aligned} \Delta u &= \frac{1}{2\pi\sigma\sqrt{2}} \\ \Delta v &= \frac{\lambda}{2\pi\sigma\sqrt{2}}. \end{aligned}$$

Multiplication confirms that the optimal uncertainty relation (4.3 on page 57) is satisfied for this family of filters:

$$\Delta x \Delta u \Delta y \Delta v = \frac{1}{16\pi^2}. \quad (\text{A.2})$$

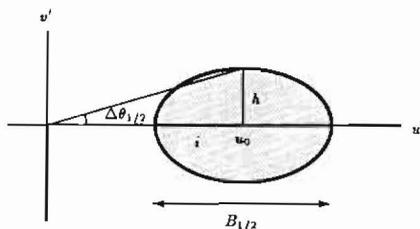


FIGURE A.2: Gabor orientation and frequency bandwidths. The diagram illustrates the frequency response of a Gabor filter, from which bandwidths are defined as indicated.

A.2.2 Orientation and Frequency Bandwidths

The frequency response of a Gabor filter is shown in Figure 4.2 on page 59: it is a displaced Gaussian. We shall use the idealised scheme shown in Figure A.2 to define the orientation selectivity $\Delta\theta_{1/2} = \arctan(h/u_0)$. Our filter has a principal frequency response at $u_0 = \omega/2\pi$ — this is the centre of the ellipse in Figure A.2. Taking the half-height bandwidth by setting $Q(u_0, h) = 1/2$ we get:

$$h = \frac{\lambda\sqrt{2\ln 2}}{2\pi\sigma}$$

and hence³

$$\tan \Delta\theta_{1/2} = \frac{\alpha\lambda}{\sigma\omega} \quad (\text{A.3})$$

where $\alpha = \sqrt{2\ln 2}$. The half-height angular frequency bandwidth may be obtained similarly from setting $Q(u_0 - i, 0) = 1/2$: $\Delta\omega_{1/2} = \alpha/\sigma$. It is more common to express this in terms of frequency octaves:

$$\begin{aligned} B_{1/2} &= \log_2 \left(\frac{\omega + \Delta\omega_{1/2}}{\omega - \Delta\omega_{1/2}} \right) \\ &= \log_2 \left(\frac{\sigma\omega + \alpha}{\sigma\omega - \alpha} \right). \end{aligned} \quad (\text{A.4})$$

If we take the bandwidths $B_{1/2}$ and $\Delta\theta_{1/2}$ as the primary characteristics, these equations may be inverted to give the kernel parameters σ and λ :

$$\begin{aligned} \sigma\omega &= \alpha\beta \\ \lambda &= \beta \tan \Delta\theta_{1/2} \end{aligned} \quad (\text{A.5})$$

where

$$\beta = \frac{2^{B_{1/2}} + 1}{2^{B_{1/2}} - 1}.$$

This parameterisation for the filter kernel q is more appropriate than (A.1) because frequency and orientation bandwidths reflect image properties and hence are of greater relevance than kernel shape.

³Our result differs from Daugman (1985) who found: $\sin \Delta\theta_{1/2} = \frac{\alpha\lambda}{\sigma\omega}$.

A.2.3 Gabor Aliasing Error

In this section, we will derive the aliasing error caused by discrete convolution of the Gabor filter $q(x, y)$ (A.1). Let us assume that the critical sampling density is perpendicular to the sinusoid, along the x' -axis (Bovik *et al.*, 1990). This holds for all reasonable filter coefficients and allows us to simplify our analysis to 1-D using the kernel $h \Leftrightarrow H$:

$$h(x) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{x^2}{2\sigma^2}\right) \exp(j\omega x)$$

$$H(u) = \exp\left[-\frac{1}{2}(2\pi u - \omega)^2 \sigma^2\right].$$

Aliasing error ξ may be taken as the proportion of filter energy that lies above the image half-sampling frequency f_s . For a uniform sampling period X , $f_s = (2X)^{-1}$. We shall assume (safely) that truncation of the filter is negligible.

$$\xi = \frac{\int_{f_s}^{\infty} H(u)H^*(u) du}{\int_0^{\infty} H(u)H^*(u) du} \quad (\text{A.6})$$

Write

$$I = \int_{f_s}^{\infty} \exp[-(2\pi u - \omega)^2 \sigma^2] du$$

$$= \int_{f_s - u_0}^{\infty} \exp[-(2\pi \zeta)^2 \sigma^2] d\zeta$$

where $u_0 = \omega/2\pi$ and $\zeta = u - u_0$. Substitute $\eta = 2\sqrt{2}\pi\sigma\zeta$:

$$I \propto \int_{-a}^{\infty} \exp(-\eta^2/2) d\eta$$

$$\propto \Phi(a)$$

where $\Phi(\cdot)$ is the inverse-Normal distribution and $a = 2\sqrt{2}\pi\sigma(u_0 - f_s)$. We may write sampling ratio $\rho = f_s/u_0$, and hence:

$$\xi = \frac{\Phi[\gamma(1 - \rho)]}{\Phi(\gamma)}$$

$$\gamma = \sigma\omega\sqrt{2}$$

but from (A.5) $\sigma\omega = \alpha\beta$ and hence⁴ $\gamma = \alpha\beta\sqrt{2}$.

If an image is sampled on a rectangular array of pixels, it is tempting to set the image sampling period X to one pixel, $X = 1$, and hence half-sampling frequency $f_s = 1/2$. However, in the worst-case direction along a diagonal, $X = \sqrt{2}$ and $f_s = \frac{1}{2\sqrt{2}}$. Taking this into account, we arrive at the result (4.9) quoted on page 60.

⁴This differs from the relation obtained by Bovik *et al.* (1990): $\gamma = \alpha^2\beta/2$.

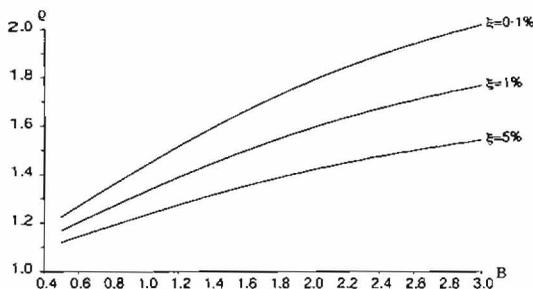


FIGURE A.3: Relation between Gabor aliasing error and sampling density. Relation between sampling ratio ρ and filter frequency bandwidth $B_{1/2}$ for fixed sampling error ξ_{\max} . The Nyquist sampling limit corresponds to $\rho = 1$. As the required level of accuracy is increased, the image must be sampled more densely.

Over- and Under-Sampling

If we are given a maximum permissible error ξ_{\max} , we may invert (4.9) to give the required sampling density τ_{\max} :

$$\tau_{\max} = \frac{\pi\gamma\sqrt{2}}{\omega[\gamma - \Phi^{-1}(\xi_{\max}\Phi(\gamma))]} \quad (\text{A.7})$$

The relationship between sampling ratio ρ and frequency bandwidth $B_{1/2}$ for fixed ξ_{\max} is shown in Figure A.3: for low-frequency filters, T large, it is possible to take advantage of $\tau_{\max} > 1$ by sub-sampling the image. For the example filter ($\Delta\theta_{1/2} = \pi/6$, $B_{1/2} = 1$) discussed in §4.1.2, we have $\tau_{\max} = 3.65$ pixels when $\xi_{\max} = 1\%$.

Sampling ratio ρ is an asymptotically increasing function of $B_{1/2}$, and to be certain that ξ_{\max} is not exceeded whatever the value of $B_{1/2}$, we may take $B_{1/2} \rightarrow \infty$ and $\beta = 1$:

$$\rho_{\max} = 1 - \frac{\Phi^{-1}[\xi_{\max}\Phi(\alpha\sqrt{2})]}{\alpha\sqrt{2}} \quad (\text{A.8})$$

For a permissible error level $\xi_{\max} = 1\%$, the maximum sampling ratio $\rho_{\max} = 1.99$, and on a rectangular pixel grid ($\tau = \sqrt{2}$), this gives $T > 5.63$ pixels.

A.2.4 Elimination of Local Image Mean

The cosine phase Gabor filter q_c (4.6 on page 58) has a small response to a uniform field (§A.2.5):

$$\iint q_c(x, y) dx dy = \exp(-\sigma^2\omega^2/2). \quad (\text{A.9})$$

Unless eliminated, this will perturb the real filter outputs for each image. Linear filter response R may be adjusted by subtracting an appropriate correction, but the effect is particularly serious for the non-linear quantities \mathcal{R} and ϕ . We apply a local "DC"

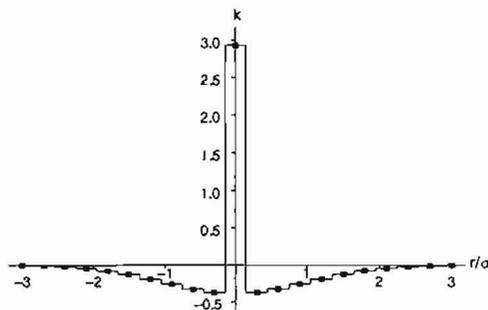


FIGURE A.4: Filter kernel used to ensure that local image mean is zero. Convolution with this zero-sum filter profile, an inverted Gaussian $g(\sigma_{DC})$ plus a δ -function, ensures that the image has negligible local mean and hence avoids perturbing Gabor filter output. As drawn here, $\sigma_{DC} = 3$ pixels.

correction to each image to subtract the local mean luminance by convolution with the profile shown in Figure A.4 (Bnr & Adelson, 1983; Heeger, 1987). Ideally, this kernel k_{DC} is a δ -function added to an inverted Gaussian:

$$k_{DC}(x, y) = \delta(x, y) - \frac{1}{2\pi\sigma_{DC}^2} \exp\left(-\frac{x^2 + y^2}{2\sigma_{DC}^2}\right) \quad (\text{A.10})$$

where we usually set $\sigma_{DC} = \sigma$, or $\sigma_{DC} = \lambda\sigma$ for $\lambda > 1$, but in practice, a discrete approximation to k_{DC} (A.10) is used. Some residual DC signal will remain because the filter k_{DC} is circular whereas the Gabor kernels are usually elliptical, but this effect is not significant.

Although direct convolution is possible, and the filter kernel k_{DC} is separable, it is most convenient to apply this correction in conjunction with the FFT algorithm (page 61). We have:

$$\mathcal{F}(I * q) = \mathcal{F}(I) \cdot \mathcal{F}(q)$$

which may be extended to write

$$\mathcal{F}[(I * k_{DC}) * q] = \mathcal{F}(I) \cdot \mathcal{F}(k_{DC}) \cdot \mathcal{F}(q).$$

The filter transform $\mathcal{F}(k_{DC}) = K_{DC}(u, v)$ may be written down:

$$K_{DC}(u, v) = 1 - \exp\left[-\frac{1}{2}(2\pi\sigma_{DC}u)^2\right] \exp\left[-\frac{1}{2}(2\pi\sigma_{DC}v)^2\right].$$

Other DC-elimination strategies are possible: Bovik *et al.* (1990) simply apply a small threshold to Gabor amplitude $A(x, y)$ while Perry and Lowe (1989) adjust their discrete mask kernels to ensure they sum to zero.

A.2.5 Gabor Response to Image Primitives

Our Profile feature-extraction algorithm (§4.5) compares observed filter responses with Gabor signatures of parameterised image primitives. In this section, we derive the

response of a Gabor filter q , parameterised as (4.6) on page 58, to a uniform field and to LINE and SINU primitives. Following Chapter 4, we make a number of definitions which are repeated here for clarity:

$$\begin{aligned}
 q(x, y) &= \frac{\lambda}{2\pi\sigma^2} \exp\left(-\frac{x'^2}{2\sigma^2}\right) \exp\left(-\frac{y'^2\lambda^2}{2\sigma^2}\right) \exp(j\omega x') & (A.11) \\
 \begin{pmatrix} x' \\ y' \end{pmatrix} &= \begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} \\
 \begin{pmatrix} \omega_x \\ \omega_y \end{pmatrix} &= \begin{pmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{pmatrix} \begin{pmatrix} \omega \\ 0 \end{pmatrix} \\
 \nu^2 &= \sin^2\theta + \lambda^2 \cos^2\theta \\
 \kappa &= (\lambda^2 - 1) \sin\theta \cos\theta \\
 t^2 &= \frac{\lambda^2 + \kappa^2}{\nu^2} \\
 R(x, y) &= \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} q(\zeta, \eta) f(x + \zeta, y + \eta) d\zeta d\eta \\
 &= R_c + jR_s & (A.12) \\
 \mathcal{R}(x, y) &= R_c^2 + R_s^2
 \end{aligned}$$

and we may express $q(x, y) = q_c + jq_s$ (A.11) in terms of the auxiliary variables defined above:

$$\begin{aligned}
 q_s(x, y) &= \frac{\lambda}{2\pi\sigma^2} \exp\left[-\frac{1}{2\sigma^2}(x^2 t^2 + 2xy\kappa + y^2 \nu^2)\right] \sin(\omega_x x + \omega_y y) \\
 q_c(x, y) &= \frac{\lambda}{2\pi\sigma^2} \exp\left[-\frac{1}{2\sigma^2}(x^2 t^2 + 2xy\kappa + y^2 \nu^2)\right] \cos(\omega_x x + \omega_y y).
 \end{aligned}$$

We shall also use the following results from contour integration:

$$\frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} \exp(-x^2/2) dx = 1 \quad (A.13)$$

$$\begin{aligned}
 \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} \exp(-x^2/2) \cos \lambda x dx &= \frac{\exp(-\lambda^2/2)}{\sqrt{2\pi}} \int_{-\infty-j\lambda}^{\infty-j\lambda} \exp(-x^2/2) dx \\
 &= \exp(-\lambda^2/2). & (A.14)
 \end{aligned}$$

Gabor Signature of Uniform Field

We take f_{DC} as the image function, where

$$f_{DC}(x, y) = c.$$

Since this function is rotationally symmetric, we may set $\theta = 0$ for convenience, and derive the Gabor signature from continuous linear convolution (A.12):

$$\begin{aligned}
 R_s &= \frac{c\lambda}{2\pi\sigma^2} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \exp\left(-\frac{x^2}{2\sigma^2}\right) \exp\left(-\frac{y^2\lambda^2}{2\sigma^2}\right) \sin \omega x dx \\
 &= 0 & (A.15)
 \end{aligned}$$

because the integrand is antisymmetric around $x = 0$, and

$$\begin{aligned} R_c &= \frac{c\lambda}{2\pi\sigma^2} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \exp\left(-\frac{x^2}{2\sigma^2}\right) \exp\left(-\frac{y^2\lambda^2}{2\sigma^2}\right) \cos\omega x \, dx \\ &= c \exp(-\omega^2\sigma^2/2) \end{aligned} \quad (\text{A.16})$$

from application of (A.13) and (A.14). We are interested in the spatial variation of the image, not its mean value, and a local correction is made during pre-processing to ensure that filter response is not perturbed (§A.2.4).

Gabor Signature of Ideal Line: LINE

For the LINE primitive, we take the image function to be an ideal line formed by an extended δ -function, as (4.23) on page 82:

$$f_{\text{LINE}}(x, y) = c\delta(x - d).$$

For convenience, we take the line to be parallel to the y -axis and consider the application of a filter tuned to angle θ , taking line offset d as the perpendicular distance from the line to the filter origin measured along the filter x -axis (so d may be negative), and setting $-\pi/2 < \theta \leq \pi/2$:

$$\begin{aligned} R_s &= \frac{c\lambda}{2\pi\sigma^2} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \exp\left[-\frac{1}{2\sigma^2}(x^2\iota^2 + 2xy\kappa + y^2\nu^2)\right] \sin(\omega_x x + \omega_y y) \delta(x - d) \, dx \, dy \\ &= \frac{c\lambda}{2\pi\sigma^2} \exp\left(-\frac{d^2\lambda^2}{2\sigma^2\nu^2}\right) \int_{-\infty}^{\infty} \exp\left[-\frac{\nu^2}{2\sigma^2}\left(y + \frac{d\kappa}{\nu^2}\right)^2\right] \sin(\omega_x d + \omega_y y) \, dy \\ &= \frac{c\lambda}{\nu\sigma\sqrt{2\pi}} \exp\left(-\frac{d^2\lambda^2}{2\sigma^2\nu^2}\right) \sin\left(\frac{\omega d\lambda^2 \cos\theta}{\nu^2}\right) \exp\left(-\frac{\omega^2\sigma^2 \sin^2\theta}{2\nu^2}\right). \end{aligned} \quad (\text{A.17})$$

Similarly,

$$\begin{aligned} R_c &= \frac{c\lambda}{2\pi\sigma^2} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \exp\left[-\frac{1}{2\sigma^2}(x^2\iota^2 + 2xy\kappa + y^2\nu^2)\right] \cos(\omega_x x + \omega_y y) \delta(x - d) \, dx \, dy \\ &= \frac{c\lambda}{\nu\sigma\sqrt{2\pi}} \exp\left(-\frac{d^2\lambda^2}{2\sigma^2\nu^2}\right) \cos\left(\frac{\omega d\lambda^2 \cos\theta}{\nu^2}\right) \exp\left(-\frac{\omega^2\sigma^2 \sin^2\theta}{2\nu^2}\right). \end{aligned} \quad (\text{A.18})$$

Gabor Signature of Infinite Sinusoid Grating: SINU

Our SINU primitive is an infinite sinusoid grating, as (4.24) on page 82:

$$f_{\text{SINU}}(x, y) = c \sin(\varpi x + \psi).$$

We take $\varpi > 0$ and $-\pi/2 < \theta \leq \pi/2$. For convenience, we take the sinusoid parallel to the y -axis:

$$\begin{aligned}
R_s &= \frac{c\lambda}{2\pi\sigma^2} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \exp\left[-\frac{1}{2\sigma^2}(x^2\iota^2 + 2xy\kappa + y^2\nu^2)\right] \\
&\quad \times \sin(\omega_x x + \omega_y y) \sin(\varpi x + \psi) \, dx \, dy \\
&= \frac{c\lambda}{4\pi\sigma^2} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \exp\left(-\frac{u^2\iota^2}{2\sigma^2}\right) \exp\left(-\frac{y^2\lambda^2}{2\sigma^2\iota^2}\right) \\
&\quad \times \left\{ -\cos[(\omega_x + \varpi)u] \cos\left(y[\omega_y - (\omega_x + \varpi)\kappa/\iota^2] + \psi\right) \right. \\
&\quad \left. + \cos[(\omega_x - \varpi)u] \cos\left(y[\omega_y - (\omega_x - \varpi)\kappa/\iota^2] - \psi\right) \right\} du \, dy \\
&= \frac{c\lambda \cos \psi}{2\iota\sigma\sqrt{2\pi}} \exp\left[-\frac{(\omega_x^2 + \varpi^2)\sigma^2}{2\iota^2}\right] \int_{-\infty}^{\infty} \exp\left(-\frac{y^2\lambda^2}{2\sigma^2\iota^2}\right) \\
&\quad \times \left\{ -\exp(-\omega_x\varpi\sigma^2/\iota^2) \cos\left(y[\omega_y - (\omega_x + \varpi)\kappa/\iota^2]\right) \right. \\
&\quad \left. + \exp(+\omega_x\varpi\sigma^2/\iota^2) \cos\left(y[\omega_y - (\omega_x - \varpi)\kappa/\iota^2]\right) \right\} du \, dy \\
&= c \cos \psi \exp\left(-\frac{\sigma^2\nu^2\varpi^2}{2\lambda^2}\right) \exp(-\sigma^2\omega^2/2) \sinh[\omega\varpi\sigma^2 \cos \theta] \quad (\text{A.19})
\end{aligned}$$

$$R_c = c \sin \psi \exp\left(-\frac{\sigma^2\nu^2\varpi^2}{2\lambda^2}\right) \exp(-\sigma^2\omega^2/2) \cosh[\omega\varpi\sigma^2 \cos \theta]. \quad (\text{A.20})$$

A.3 Alternative Gabor Feature Estimators

A principled approach to the problem of how to pool measurements from many Gabor channels to form a single image description was proposed in Chapter 4. Our objectives, established in §4.5.1, are to express measured responses in many Gabor channels in a compact and convenient form by matching them to Gabor signatures formed by parameterised image primitives. This matching problem is over-constrained but the measurements may be noisy, and analytic inversion of the response equations, (4-25–4-28) or (4-29–4-32) on page 83, is not appropriate. Least-squares feature estimation is desirable in principle but complex in practice (§4.5.3), and in §4.6 we set out our Profile and Resultant feature-extraction methods, which both exploit response moments. Whilst these both generate acceptable features (§4.6.4), and are successfully used in Chapter 6 in conjunction with our hybrid models, some compromises of our original objectives were involved. In this section, we briefly explore some alternative feature-extraction strategies, and explain why these were rejected in favour of the Profile and Resultant methods.

Notation

Following Chapter 4 (page 69), we specify sets of filters by: sinusoid period T (pixels), half-height octave spatial-frequency bandwidth $B_{1/2}$, spacing between orientations Θ , and half-height orientation bandwidth $\Delta\theta_{1/2}$. From these may be derived the sinusoid frequency $\omega = 2\pi/T$ (1/pixel units), space constant of Gaussian window σ (pixel units), and window aspect ratio λ , as described in §4.1. Individual filters are specified additionally by their orientation tuning θ , which we restrict to the range $0 \leq \theta < \pi$ because filters $q(\theta)$ and $q(\theta + \pi)$ are equivalent up to a change of sign (4-6 on page 58). Channel outputs were combined from several orientations but a single frequency. Additionally, we employ the notation from page 82 to specify image primitives:

LINE: contrast c (gray-level units), line offset d (pixels), and orientation θ_0 (radians);

SINU: contrast c (gray-level units), grating phase ψ (radians), angular frequency ω (radians per pixel), and orientation θ_0 (radians).

A.3.1 Estimation of LINE Features

Tabulation of One Parameter

The Gabor signature of the LINE primitive is parameterised by three “features”: orientation θ_0 , contrast c , and line offset d (4-25–4-28 on page 83). Of these, variation with θ_0 takes the most complicated form and holds out least prospect for direct evaluation, and consequently it is sensible to seek alternative estimators for this feature first. One alternative approach is to estimate orientation by partial tabulation of possible values before seeking a local best fit for the remaining pair of parameters. This is tantamount to quantising orientation, say to units of 5° , and performing exhaustive search for all allowable values. Let us assume that terms used to compute the model signature $\mathcal{R}_{\text{LINE}}$

are tabulated at the orientations $\{\alpha_k\}$. Local least-squares estimation (§4.5.3) may be used for each possible orientation α_k , comparing residual errors ε_k to select the best combination:

$$\varepsilon_k = \|\mathbf{M} - \mathbf{R}(\alpha_k)\|^2 \quad (\text{A.21})$$

for measured response vector \mathbf{M} and model signature $\mathbf{R}(\alpha_k)$, and setting

$$\frac{\partial \varepsilon_k}{\partial c} = \frac{\partial \varepsilon_k}{\partial d} = 0 \quad (\text{A.22})$$

to get \hat{c}_k , \hat{d}_k and $\varepsilon_{k,\min}$. We then select the orientation α_ℓ which minimises $\varepsilon_{k,\min}$:

$$\hat{\theta}_0 = \alpha_\ell, \text{ if } \varepsilon_{\ell,\min} = \min_k(\varepsilon_{k,\min}). \quad (\text{A.23})$$

Unfortunately, (A.22) cannot be solved easily because the function $R(d)$ is too complicated (4.25–4.26 on page 83).

Gabor phase ϕ (4.28 on page 83) varies more simply with line offset d , and we may consider an estimator based on the phase of the observed response, again assuming tabulation of θ :⁵

$$\varepsilon_k \stackrel{?}{=} \sum_i \left(\frac{M_{s,i}}{M_{c,i}} - \tan[\phi(\alpha_k - \theta_i)] \right)^2$$

but this is also difficult to solve for \hat{d}_k , and it is tempting to take the argument one stage further to write

$$\varepsilon_k \stackrel{?}{=} \|\mathbf{P} - \phi\|^2$$

where $P_i = \arctan(M_{s,i}/M_{c,i})$, and hence

$$\hat{d}_k \stackrel{?}{=} \frac{1}{\omega\lambda^2} \frac{\sum_i P_i \cos(\alpha_k - \theta_i)/\nu_i^2}{\sum_i \cos^2(\alpha_k - \theta_i)/\nu_i^4}.$$

Unfortunately, this expression is erroneous because no account has been taken of the periodicity of \arctan , and we should replace the above definition of P_i by:

$$P_i = \arctan(M_{s,i}/M_{c,i}) \pm m\pi, \quad m = 0, 1, \dots$$

but this creates many additional possible solutions which must be investigated. Periodicity is a problem only for moderate line offset, $|d| > T/2$, but it is quite possible for this range to be exceeded in practice (see Figure 4.18 on page 86), and significant errors will result unless the spatial frequency bandwidth $B_{1/2}$ (4.8 on page 59) is set very high, implying the Gabor envelope has limited spatial extent (σ small) and restricting the effective range of possible solutions for d .

Instead, it is possible to obtain an estimate from the Gabor energy \mathcal{R} (4.27) by writing:

$$\ln \mathcal{R} = \ln \left(\frac{c^2 \lambda^2}{2\pi \nu^2 \sigma^2} \right) - \frac{\lambda^2 d^2}{\sigma^2 \nu^2} - \frac{\omega^2 \sigma^2 \sin^2(\theta_0 - \theta)}{\nu^2}. \quad (\text{A.24})$$

⁵This estimator is incorrect because our confidence in the observed phase M_s/M_c is not the same for all measurements: see the treatment in (A.25).

We may now write a least-squares estimator in terms of $\{\ln \mathcal{M}_i\}$, but because these are derived quantities, we can no longer assume that all errors are of the same magnitude — as with (4.41) on page 89, we normalise each term by its estimated standard deviation:

$$\begin{aligned}\Delta(\ln \mathcal{M}) &= \Delta \mathcal{M} / \mathcal{M} \\ &= 2\Delta_M \mathcal{M}^{-1/2}\end{aligned}$$

where $\Delta M_s = \Delta M_c = \Delta_M$. Hence:

$$\varepsilon_k = \sum_i (\ln \mathcal{M}_i - \ln \mathcal{R}_i)^2 \mathcal{M}_i \quad (\text{A.25})$$

which gives:

$$\begin{aligned}\hat{d}_k^2 &= \frac{(\sum_i \mathcal{M}_i) [\sum_\ell \mathcal{M}_\ell B_\ell (A_\ell - \ln \mathcal{M}_\ell)] - (\sum_i \mathcal{M}_i B_i) [\sum_\ell \mathcal{M}_\ell (A_\ell - \ln \mathcal{M}_\ell)]}{(\sum_i \mathcal{M}_i B_i^2) (\sum_i \mathcal{M}_i) - (\sum_i \mathcal{M}_i B_i)^2} \\ \hat{c}_k^2 &= \frac{\sum_i [\mathcal{M}_i (\ln \mathcal{M}_i - A_i + B_i \hat{d}_k^2)]}{\sum_i \mathcal{M}_i}\end{aligned}$$

where summation for i and ℓ extends over all channels, and:

$$\begin{aligned}A_i &= \ln \left(\frac{\lambda^2}{2\pi \nu_i^2 \sigma^2} \right) - \frac{\omega^2 \sigma^2 \sin^2(\alpha_k - \theta_i)}{\nu_i^2} \\ B_i &= \frac{\lambda^2}{\sigma^2 \nu_i^2} \\ \nu_i^2 &= \sin^2(\alpha_k - \theta_i) + \lambda^2 \cos^2(\alpha_k - \theta_i)\end{aligned}$$

plus the trivial solution $\hat{d}_k^2 = 0$. In principle, these estimators (\hat{d}^2 and \hat{c}^2) are fully functional, and we have tested the results on several images. The main effect on Gabor energy \mathcal{R} of increasing line offset d is a scaling by approximately: $\exp(-d^2/\sigma^2)$; the above estimators depend on this factor varying slightly with angle as: $\exp(-\lambda^2 d^2/\sigma^2 \nu^2)$. The distinction between these two terms is illustrated in Figure A.5: for typical values of filter eccentricity λ , it is too small to give reliable estimators for c^2 and d^2 .

Tabulation of Two Parameters

We found above that the variation of the Gabor signature R_{LINE} was too complicated to permit accurate parameter estimation by simple methods, even after obtaining orientation by partial tabulation, and that Gabor energy \mathcal{R} and phase ϕ were not sufficiently sensitive to the required parameters to form reliable estimators. We now return to the channel response R_{LINE} , tabulating values for line offset d to simplify the residual estimation problem further. This may be done in conjunction with tabulation to obtain $\hat{\theta}_0$ (although this will be very slow), or with the moment estimator (§4.6.1).

Given values for θ_0 and d , it is straightforward to write down a local estimator for the contrast c :

$$\hat{c} = \frac{\sigma \sqrt{2\pi}}{\lambda} \frac{\sum_i \left(\exp \left(-\frac{\lambda^2 d^2}{2\sigma^2 \nu_i^2} \right) \exp \left(-\frac{\sigma^2 \omega^2 \sin^2(\theta_0 - \theta_i)}{2\nu_i^2} \right) \times \left[M_c \cos \left(\frac{\omega d \lambda^2 \cos(\theta_0 - \theta_i)}{\nu_i^2} \right) + M_s \sin \left(\frac{\omega d \lambda^2 \cos(\theta_0 - \theta_i)}{\nu_i^2} \right) \right] / \nu_i}{\sum_i \exp \left(-\frac{\lambda^2 d^2}{\sigma^2 \nu_i^2} \right) \exp \left(-\frac{\sigma^2 \omega^2 \sin^2(\theta_0 - \theta_i)}{\nu_i^2} \right) / \nu_i^2}$$

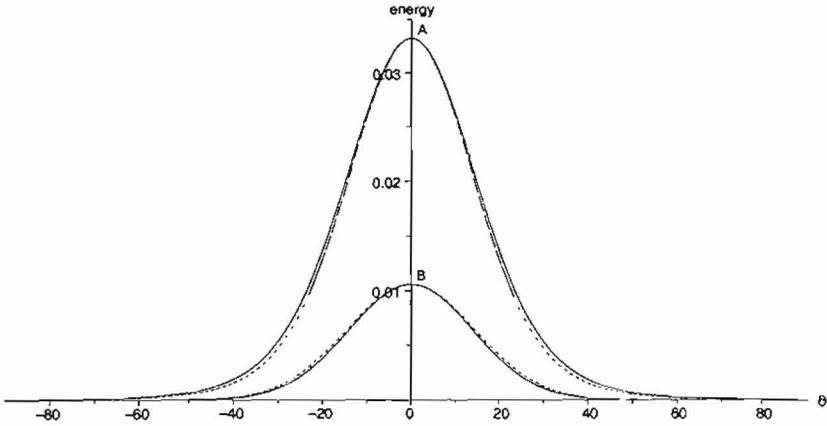


FIGURE A.5: Gabor energy variation for LINE primitive. We postulated an estimation method for LINE features based on the difference between dotted and solid lines, shown as a function of orientation tuning ϑ (degrees), but this was too small in practice to extract reliable parameters. (a) $T = 3.5$ pixels, $B_{1/2} = 1.5$, $\Delta\theta_{1/2} = \pi/8$, $d = 1.75$ pixels; (b) $T = 4$ pixels, $B_{1/2} = 1$, $\Delta\theta_{1/2} = \pi/8$, $d = 3$ pixels, using the notation described on page 259.

where ν_i is given by (4.37) on page 87. This estimation method was implemented in conjunction with the moment estimator, and tested on some simple images. Although the observed signatures $\mathbf{M}(x, y)$ were modelled closely, and the residual error ε was small, the estimates obtained for contrast c and line offset d were often hopelessly wrong: small changes in d cause rapid oscillations in the match error ε , with the global minimum error being only slightly smaller than many local minima into which the inflexible search algorithm readily falls. This problem is a combination of those experienced when treating phase and amplitude envelopes separately:

$$R = \frac{c\lambda}{\nu\sigma\sqrt{2\pi}} \underbrace{\exp\left(-\frac{\lambda^2 d^2}{2\sigma^2 \nu^2}\right)}_A \exp\left[-\frac{\omega^2 \sigma^2 \sin^2(\theta_0 - \theta)}{2\nu^2}\right] \underbrace{\exp\left[j\frac{\omega d \lambda^2 \cos(\theta_0 - \theta)}{\nu^2}\right]}_B$$

term A is too insensitive to estimate d because $\nu(\theta)$ varies only weakly; while the trigonometric term B oscillates rapidly to give many false matches. We may estimate the density of local minima to see whether a finer tabulation of d would help: let us assume that the principal component of the observed response vector \mathbf{M} lies at the correct orientation, $\vartheta = 0$. We may fit this term exactly by:

$$\begin{aligned} \hat{d}_i &= d_0 \pm iT/2 \\ \hat{c}_i &= c_0 (-1)^{\pm i} \exp\left(\frac{d_1^2 - d_0^2}{2\sigma^2}\right) \end{aligned}$$

where $i = 0, 1, \dots$. Only filters away from the match angle will be able to differentiate between these solutions, similar to the situation illustrated in Figure A.5, but unfor-

unately these have the weakest responses and hence are most noise prone. This is a serious flaw with the tabulation procedure, and cannot easily be overcome.

Circular Filters

The position is simpler for a circular Gabor filter ($\lambda = 1$), which has energy response \mathcal{R}_\circ :

$$\mathcal{R}_\circ = \frac{c^2}{2\pi\sigma^2} \exp\left(-\frac{d^2}{2\sigma^2}\right) \exp[-\omega^2\sigma^2 \sin^2(\theta_0 - \theta)]. \quad (\text{A-26})$$

If we are content to define a “feature strength” parameter $s = c^2 \exp(-d^2/\sigma^2)$, \hat{s} may easily be obtained from the response vector \mathbf{M} , assuming that θ_0 is known or estimated independently. In practice, setting $\lambda = 1$ imposes too many restrictions on the orientation bandwidth $\Delta\theta_{1/2}$, and it is in any case unclear what advantages the above procedure would offer over that described on page 89.

Filtering at Multiple Scales

One final avenue is to pool output from filters at several scales. Let us consider two responses, at filter frequencies ω and Ω :

$$\begin{aligned} \mathcal{R}_\omega &= \frac{c^2\lambda^2}{2\pi\sigma^2\nu^2} \exp\left(-\frac{\lambda^2 d^2}{\sigma^2\nu^2}\right) \exp\left(-\frac{\omega^2\sigma^2 \sin^2(\theta_0 - \theta)}{\nu^2}\right) \\ \mathcal{R}_\Omega &= \frac{c^2\Lambda^2}{2\pi\Sigma^2 N^2} \exp\left(-\frac{\Lambda^2 d^2}{\Sigma^2 N^2}\right) \exp\left(-\frac{\Omega^2\Sigma^2 \sin^2(\theta_0 - \theta)}{N^2}\right) \end{aligned} \quad (\text{A-27})$$

where

$$\begin{aligned} \nu^2 &= \sin^2(\theta_0 - \theta) + \lambda^2 \cos^2(\theta_0 - \theta) \\ N^2 &= \sin^2(\theta_0 - \theta) + \Lambda^2 \cos^2(\theta_0 - \theta). \end{aligned}$$

If filters at the same angles $\{\theta_i\}$ are used to make both sets of measurements, and $\Delta\theta_{1/2}$ and $B_{1/2}$ are also maintained, we have: $\Lambda = \lambda$, $N^2 = \nu^2$ and $\Sigma\Omega = \sigma\omega$. Hence, from (A-27):

$$\frac{\sigma^2\mathcal{R}_\omega}{\Sigma^2\mathcal{R}_\Omega} = \exp\left[\frac{\lambda^2 d^2}{\nu^2} \left(\frac{1}{\Sigma^2} - \frac{1}{\sigma^2}\right)\right].$$

This could be used as the basis for a multi-scale estimator for line offset d , but has not currently been implemented. Further discussion appears in §8.2.

A.3.2 Estimation of SINU Features

As with the LINE primitive, direct least-squares estimates of SINU features are not readily available (§4.5.3), and instead we consider approximate methods. Exact variation of the Gabor signature R_{SINU} with filter orientation θ is complicated (4-29-4-32 on page 83), and consequently we shall assume that the grating orientation θ_0 has been estimated either by the moment method (§4.6.1) or by tabulation of partial results and exhaustive search. In this section, we consider approaches to determine the remaining SINU features: grating frequency ϖ , contrast c , and phase ψ .

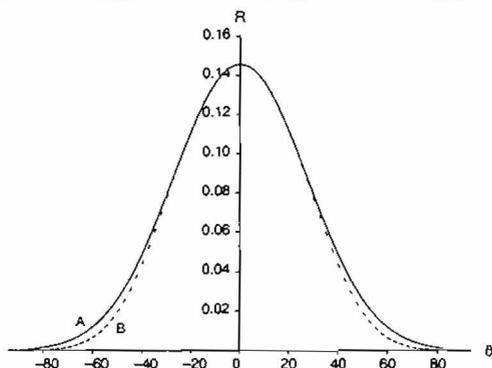


FIGURE A.6: Gabor response variation for SINU primitive. We postulated an estimation method for SINU features based on the difference between dotted and solid lines but this was too small in practice to extract reliable parameters. (a) sine phase response R_s for frequency ratio $\omega = 0.56\omega_0$; (b) scaled response for equal filter and grating frequencies. Filter parameters: ($T = 5$ pixels, $B_{1/2} = 1$, $\Delta\theta_{1/2} = \pi/6$; $\sigma = 2.8$ pixels, $\psi = 0$), using the notation described on page 259.

Sinusoid Frequency

The major effect of a change in the frequency of the image sinusoid ω is to scale the channel response R_{SINU} by: $\exp[-\sigma^2(\omega - \omega_0)^2/2]$; and hence Gabor energy \mathcal{R} by: $\exp[-\sigma^2(\omega - \omega_0)^2]$. The difference between a pure scaling and the perturbing effect alone is shown in Figure A.6: it is insufficient in practice to establish a reliable estimator $\hat{\omega}$ from a single set of observations. Consequently, our approach is to abandon direct estimation of the grating frequency ω , and set $\omega = \omega_0$. When this is substantially in error, the apparent contrast will be small, because it decays roughly as: $c^2 \propto \exp[-\sigma^2(\omega - \omega_0)^2]$.

Phase Estimator

From (4.32) on page 83, the phase of the filter response ϕ is given by:

$$\tan \phi = \cot \psi \tanh(\omega \omega_0 \sigma^2 \cos(\theta_0 - \theta)).$$

In most cases, we have $\omega \omega_0 \sigma^2 > 1$, and except for $\theta_0 - \theta \approx \pm \pi/2$ we may write:

$$\tan \phi \approx \cot \psi \tag{A.28}$$

(see Figure 4.19b on page 86). This allows an informal estimate of ψ , but this is not done at present.

Contrast Estimator

When we assume grating and filter frequencies to be equal ($\varpi = \omega$), the model energy response \mathcal{R} is given by:

$$\mathcal{R}(\varpi = \omega) = c^2 \exp \left[-\frac{\sigma^2 \omega^2 (\sin^2 \theta + \lambda^2 \cos^2 \theta)}{\lambda^2} \right] \exp(-\sigma^2 \omega^2) \\ \times [\sin^2 \psi + \sinh^2(\sigma^2 \omega^2 \cos \theta)].$$

In practice, $\sin^2 \psi \ll \sinh^2(\omega^2 \sigma^2 \cos \theta)$, and we may drop this term. The contrast may then be estimated using the weighted least-squares procedure described in §4.6.2, leading to the Profile contrast estimator (4.42) on page 89.

Multiple Frequencies

If we take c_{ϖ} as the true contrast, measured by a filter tuned to the grating frequency, $\omega = \varpi$, then apparent contrast c^2 varies approximately as:

$$c^2(\omega) \approx c_{\varpi}^2 \exp[-\sigma^2(\varpi - \omega)^2].$$

By taking two or more filter frequencies, we may estimate c_{ϖ}^2 and ϖ . The validity of these expressions depends on the approximation:

$$\sinh^2(\omega \varpi \sigma^2) \approx \frac{1}{4} \exp(2\omega \varpi \sigma^2).$$

Further errors will be introduced if the image actually contains structure at several scales.

A.4 Marginal Densities of Proposed Gabor–Markov Models

In order to diminish the disparity between our Gabor features and the auto-normal model, we proposed transformations corresponding to specific Gabor–Markov models (§6.3). The purpose of this Appendix is to demonstrate that these proposed models form appropriate representations for our orientation feature θ . We adopt the notation of Chapter 6.

A.4.1 Orthogonal Normal Fields Model

In §6.3.1, we proposed that an angular field θ could be modelled by two auto-normal MRFs p and q , according to the transform:

$$\theta = \arctan(p/q) \quad (\text{A.29})$$

for $-\pi < \theta \leq \pi$, where p and q are normal and independent fields. In this section, we derive the resulting marginal density function $f_\theta(\theta)$.

Let us introduce an auxiliary variable r , and express the new distribution $f_{r,\theta}$ in terms of the known density function $f_{p,q}$:

$$\begin{aligned} r &= \sqrt{p^2 + q^2} & (\text{A.30}) \\ f_{r,\theta}(r, \theta) &= \frac{f_{p,q}(r \sin \theta, r \cos \theta)}{|\mathbf{J}(r \sin \theta, r \cos \theta)|} \\ &= r f_{p,q}(r \sin \theta, r \cos \theta) \\ f_\theta(\theta) &= \int_0^\infty r f_{p,q}(r \sin \theta, r \cos \theta) dr & (\text{A.31}) \end{aligned}$$

where $\mathbf{J}(\cdot)$ is the Jacobian matrix (Papoulis, 1991), and p and q are assumed normal and independent but not necessarily identically distributed, with means μ_p and μ_q .

$$f_{p,q}(p, q) = \frac{1}{2\pi\sigma_p\sigma_q} \exp\left[-\frac{1}{2\sigma_p^2}(p - \mu_p)^2 - \frac{1}{2\sigma_q^2}(q - \mu_q)^2\right] \quad (\text{A.32})$$

Write $\rho = \sigma_p/\sigma_q$.

$$\begin{aligned} f_\theta(\theta) &= \frac{1}{2\pi\sigma_p\sigma_q} \int_0^\infty r \exp\left\{-\frac{1}{2}\left[r^2 \frac{\rho^{-1} \sin^2 \theta + \rho \cos^2 \theta}{\sigma_p\sigma_q} - 2r \frac{\mu_p \rho^{-1} \sin \theta + \mu_q \rho \cos \theta}{\sigma_p\sigma_q} \right. \right. \\ &\quad \left. \left. + \frac{\mu_p^2/\rho + \mu_q^2 \rho}{\sigma_p\sigma_q}\right]\right\} dr \end{aligned}$$

Set

$$\begin{aligned} u &= r \sqrt{\frac{\rho^{-1} \sin^2 \theta + \rho \cos^2 \theta}{\sigma_p\sigma_q}} \\ \varepsilon &= \frac{\mu_p \rho^{-1} \sin \theta + \mu_q \rho \cos \theta}{\sqrt{\sigma_p\sigma_q (\rho^{-1} \sin^2 \theta + \rho \cos^2 \theta)}} \end{aligned}$$

and complete the square.

$$f_{\theta}(\theta) = \frac{1}{2\pi (\rho^{-1} \sin^2 \theta + \rho \cos^2 \theta)} \exp\left(-\frac{\mu_p^2/\rho + \mu_q^2\rho}{2\sigma_p\sigma_q}\right) \exp(\varepsilon^2/2) \\ \times \int_0^{\infty} u \exp\left[-\frac{(u-\varepsilon)^2}{2}\right] du$$

The integral is now straightforward:

$$f_{\theta}(\theta) = \frac{1}{2\pi (\rho^{-1} \sin^2 \theta + \rho \cos^2 \theta)} \exp\left(-\frac{\mu_p^2/\rho + \mu_q^2\rho}{2\sigma_p\sigma_q}\right) \left[1 + \varepsilon\sqrt{2\pi} \exp(\varepsilon^2/2) \Phi(\varepsilon)\right].$$

This may be simplified a little by writing

$$\tan \psi = \frac{\mu_p/\rho}{\mu_q\rho} \\ \beta^2 = \frac{\mu_p^2/\rho^2 + \mu_q^2\rho^2}{\sigma_p\sigma_q} \\ \varepsilon = \beta \frac{\cos(\theta - \psi)}{\sqrt{\rho^{-1} \sin^2 \theta + \rho \cos^2 \theta}}$$

hence

$$f_{\theta}(\theta) = \frac{1}{2\pi (\rho^{-1} \sin^2 \theta + \rho \cos^2 \theta)} \exp\left[-\frac{\beta^2}{2} \left(\frac{\cos^2 \psi}{\rho} + \rho \sin^2 \psi\right)\right] \\ \times \left[1 + \varepsilon\sqrt{2\pi} \exp(\varepsilon^2/2) \Phi(\varepsilon)\right]. \quad (\text{A.33})$$

Equal Model Variances

In practice, variances of the two orthogonal components are similar ($\sigma_p^2 \approx \sigma_q^2$), and we usually have $\rho \approx 1$. We shall assume that $\rho = 1$ from now on, and set $\sigma_p = \sigma_q = \sigma$. Hence:

$$f'_{\theta}(\theta) = \frac{1}{2\pi} \exp(-\beta^2/2) \left[1 + \beta\kappa\sqrt{2\pi} \exp(\beta^2\kappa^2/2) \Phi(\beta\kappa)\right] \quad (\text{A.34})$$

where

$$\beta^2 = \frac{\mu_p^2 + \mu_q^2}{\sigma^2} \\ \tan \psi = \mu_p/\mu_q \\ \kappa = \cos(\theta - \psi). \quad (\text{A.35})$$

A plot of $f'_{\theta}(\theta)$ is shown in Figure A.7. Although agreement with angular field data must be verified empirically, this function possesses appropriate characteristics: it is unimodal, periodic, smoothly varying, and symmetrical about the mean.

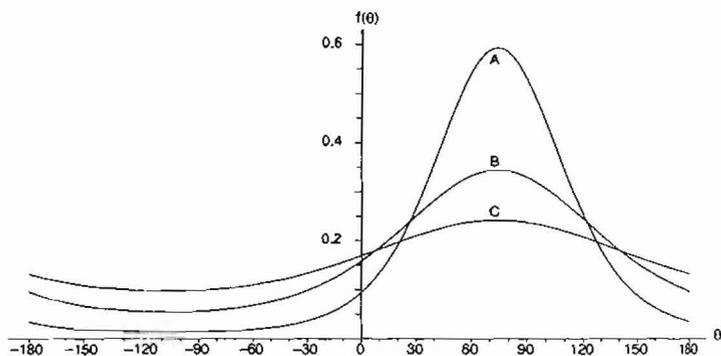


FIGURE A.7: Marginal density for orthogonal-normal model. Plots of $f_{\theta}(\theta)$ with mean $\psi = 74^{\circ}$, derived from orthogonal normal components with equal variances σ^2 . (a) $\sigma = 1/2$ pixel; (b) $\sigma = 1$ pixel; (c) $\sigma = 2$ pixels. In all cases, the distribution for θ (degrees) is symmetrical about the mean.

Mean of Angular Distribution

The mean of the angular distribution may be evaluated by integrating around a unit circle.

$$\begin{aligned} \mu &= (\mu_s, \mu_c)^T \\ &= \int_{-\pi}^{\pi} (\sin \theta, \cos \theta)^T f_{\theta}(\theta) d\theta \\ \mathcal{E}\{\theta\} &= \arctan(\mu_s/\mu_c) \end{aligned} \quad (\text{A-36})$$

The integral is only possible in the case $\rho = 1$, when

$$\mathcal{E}\{\theta\} = \psi$$

from the symmetry of $f'_{\theta}(\theta)$ about $\theta = \psi$. This is convenient, as the mean of the angular distribution is easily expressed as a function of the two normal distributions: $\mathcal{E}\{\theta\} = \arctan(\mu_p/\mu_q)$.

Note that when $\mu_p = \mu_q = 0$, ψ is not defined. The distribution then reduces to: $f'_{\theta}(\theta) = \frac{1}{2\pi}$; and $\tan \theta$ has a zero-mean Cauchy density with parameter ρ .

A.4.2 Composite Feature Model

The composite-feature model (§6.3.2) is effectively the same as (A-29) and (A-30) except that our "auxiliary variable" $\tau = c$ is now taken to be the contrast feature. From the definition of \tilde{c}^2 (§4.6) and the linearity of Gabor filtering, this form is approximately consistent with normally-distributed pixel data. When $x, y \sim G(0, \sigma^2)$, c assumes a Rayleigh density.

A.5 Properties of Modified Textures

In our discussion of our MRF texture-classification results in Chapter 7, some consequences of image corruption and sampling were assumed without proper justification. More background to these effects is presented in this Appendix, and the required results are derived.

A.5.1 Effect of Image Corruption on MRF Likelihood and Variance Terms

We examined the influence of image noise and blur on MRF likelihood in order to understand observed trends in classifier robustness (§§7.2-7.3). Three terms are important in evaluating the level of distortion of a corrupted image: the pixel variance $V(x) = \text{var } x$; the MRF predictor-error variance $V(u) = \text{var } u$; and the image-dependent penalty term of the true log-likelihood function L_X (§7.2.2):

$$\begin{aligned} L_X(x) &= \frac{x^T \mathbf{B} x}{2\sigma^2} \\ &= \sum_{i \in \mathcal{L}} \frac{|X_i|^2 (1 - B_i)}{2\sigma^2} \end{aligned} \quad (\text{A.37})$$

where summation for i extends over the dimensions of the image lattice \mathcal{L} . Our notation follows Chapters 5-7: $x = \mathbf{X}$ represents the raster-ordered image data; \mathbf{B} is a block-circulant matrix formed by rotating the MRF neighbourhood coefficients β ; \mathbf{B} is the cosine transform of β ; and σ^2 is the MRF variance. The pixel variance $V(x)$ is a property of the image alone whereas the penalty term L_X and predictor-error variance $V(u)$ also depend on the MRF parameter set \mathcal{S}_x . We shall assume for simplicity that the parameter set describes the uncorrupted data perfectly.

Usually, the MRF predictor-error variance is defined in the MRF parameter set: $V(u) = \sigma_u^2$. This correspondence may be verified:

$$\begin{aligned} u_i &= x_i - \hat{x}_i \\ &= x_i * (\delta - \beta). \end{aligned}$$

Using Parseval's theorem, $\|u\|^2 = \|U\|^2 = \sum_{i \in \mathcal{L}} |X_i(1 - B_i)|^2$. However, the expected form of the power spectrum $|X_i|^2$ is assumed known (§5.1.5):

$$|X_i|^2 = \frac{\sigma^2}{1 - B_i}$$

and hence

$$\mathcal{E}\{\|u\|^2\} = \sigma^2 \sum_{i \in \mathcal{L}} (1 - B_i)$$

but B_i has a cosine variation (5-63 on page 114), and hence sums to zero, giving $V_o(u) = \sigma^2$ as expected. Similarly,

$$V_o(x) = \frac{1}{MN} \sum_{i \in \mathcal{L}} \frac{\sigma^2}{1 - B_i} \quad (\text{A.38})$$

where the image has size $M \times N$. We may also substitute for $|X_i|^2$ in L_X :

$$\begin{aligned} \mathcal{E}\{L_{o,x}\} &= \sum_{i \in \mathcal{C}} \frac{\sigma^2(1 - B_i)}{(1 - B_i)2\sigma^2} \\ &= MN/2. \end{aligned} \quad (\text{A-39})$$

We may now examine how these terms are affected when the image is corrupted.

Effect of Image Blur

Following §6.4.3, we shall model image blur by convolution with a Gaussian kernel $g(b)$. Let us take an ideal image x and blur it to give the new image $x_i * g_i(b)$. Substituting the new spectrum $|X_i|^2 G_i^2(b)$ for $|X_i|^2$ above gives:

$$\begin{aligned} V_b(u) &= \frac{\sigma^2}{MN} \sum_{i \in \mathcal{C}} (1 - B_i) G_i^2(b) \\ &= V_o(u) \sum_{i \in \mathcal{C}} \frac{(1 - B_i) G_i^2(b)}{MN} \\ V_b(x) &= \frac{\sigma^2}{MN} \sum_{i \in \mathcal{C}} \frac{G_i^2(b)}{1 - B_i} \\ &= V_o(x) \frac{\sum_{i \in \mathcal{C}} G_i^2(b)/(1 - B_i)}{\sum_{i \in \mathcal{C}} 1/(1 - B_i)} \\ L_{b,x} &= \frac{1}{2} \sum_{i \in \mathcal{C}} G_i^2(b) \\ &= L_{o,x} \sum_{i \in \mathcal{C}} \frac{G_i^2(b)}{MN}. \end{aligned} \quad (\text{A-40})$$

Ignoring discretisation, we may approximate this last summation Σ by the corresponding integral:

$$\begin{aligned} \Sigma &\approx \left[\int_{-1}^1 \exp(-4\pi^2 \alpha^2 b^2) d\alpha \right]^2 \\ &\approx \frac{1}{4\pi b^2} [2\Phi(\sqrt{8}\pi b) - 1]^2. \end{aligned}$$

For blur parameter b sufficiently large, $\Phi(\sqrt{8}\pi b) \approx 1$ and $\Sigma \approx (4\pi b^2)^{-1}$, and hence: $L_{b,x} \approx L_{o,x}(4\pi b^2)^{-1}$. All three terms in (A-40) are smaller than for the original image.

Effect of Additive Noise

Blur is deterministic and hence its effect may be predicted exactly. In the case of random noise, we shall instead analyse the *expected* effect. Following §6.4.3, we model image noise as white, Gaussian, independent and additive. White noise has a flat spectrum, and we expect the modified power spectrum to be increased uniformly in proportion to

the noise variance. Let us add noise of variance ρ^2 to the ideal image to form $\mathbf{x} + \rho\mathbf{v}$, and substitute the new spectrum $|X_i|^2 + \rho^2$ for $|X_i|^2$ above:

$$V_n(u) = \frac{\sigma^2}{MN} \sum_{i \in \mathcal{C}} (1 - B_i) + \rho^2 (1 - B_i)^2.$$

We again appeal to the periodicity of B_i to write $\sum_{i \in \mathcal{C}} B_i = 0$. In the case of $\sum_{i \in \mathcal{C}} (1 - B_i)^2$, all cross terms sum to zero, and \cos^2 terms to $MN/2$.

$$\begin{aligned} V_b(u) &= \sigma^2 + \rho^2 \left(1 + 2 \sum_{i \in \mathcal{M}^+} \beta_i^2 \right) \\ &= V_o(u) \left[1 + \frac{\rho^2 (1 + 2 \|\beta\|^2)}{\sigma^2} \right] \\ V_n(x) &= \rho^2 + \frac{\sigma^2}{MN} \sum_{i \in \mathcal{C}} \frac{1}{1 - B_i} \\ &= V_o(x) \left(1 + \frac{\rho^2 MN}{\sigma^2 \sum_{i \in \mathcal{C}} 1/(1 - B_i)} \right) \\ L_{n,x} &= (1 + \rho^2/\sigma^2) MN/2 \\ &= L_{o,x} (1 + \rho^2/\sigma^2) \end{aligned} \tag{A.41}$$

All three terms are larger than for the uncorrupted image.

A.5.2 Frequency Analysis of Image Sampling

In this section, we establish some results which help to support our explanation for the remarkably beneficial effects of sampling (§7.4). These mostly involve the spatial frequency domain and are given for one-dimensional signals in order to enhance clarity. The spectrum of a continuous signal may be obtained by the Fourier integral:

$$F(\omega) = \int_{-\infty}^{\infty} f(x) \exp(-j\omega x) dx$$

and the discrete transform may be expressed as a matrix, \mathcal{F} . The basic result we need to invoke (without proof) is that convolution in the spatial domain corresponds to multiplication in the frequency domain and *vice versa* (Rosenfeld & Kak, 1982).

$$\mathcal{F}(a * b) = \mathcal{F}(a)\mathcal{F}(b) \tag{A.42}$$

This can be shown algebraically.

Ideal spatial sampling consists of multiplication of the signal by a regularly-spaced train of unit impulses. From (A.42), this is equivalent to convolution by a train of δ -functions in the frequency domain, forming many translated copies of the spectrum (Figure A.8). Sampling is irreversible when adjacent copies of the spectrum overlap because the extent to which they reinforce or cancel is unknown, but otherwise full recovery of the original signal is possible. The sampling theorem states that a continuous signal multiplied by an ideal sampling function may be exactly recovered if the original spectrum is contained within a bandwidth of less than half the sampling rate (Bateman



FIGURE A.8: Effect of temporal sampling on the frequency spectrum. Illustration of the effect of sampling a one-dimensional signal $x(t)$ in the temporal domain with a regular period T . (a) the signal $x(t)$ is effectively multiplied by a δ -function at the sampling points. (b) the repeat period of the spectrum $X(f)$ is related to the temporal sampling period.

& Yates, 1988). Sampling more frequently than this poses no difficulty but will clearly not generate the most compact representation. A discrete Fourier spectrum is sampled in the spectral domain, effectively extending the spatial signal with a repeat period.

Recovery of the full signal from the sampled spectrum is achieved by truncating it to its principal period, band-limiting it to half the sampling frequency. Multiplication by a rectangular window in the spectral domain is equivalent to convolution with a sinc function in the spatial domain. If the original signal was sampled at an insufficient rate, adjacent spectra overlap and hence part of the signal is shifted to a new spatial frequency, causing aliasing error (Figure 7.18 on page 206).

Blur imposes a Gaussian envelope on the spectrum, attenuating high frequencies (Figure A.9a). For a moderate degree of blur, aliasing still occurs and tends to produce a more even spectrum in the sampled image, but heavier blur effectively band-limits the image at a lower frequency, rendering aliasing insignificant.

Smooth-sampled images are windowed by unweighted averaging, which attenuates spatial frequency according to the sinc curve (Figure A.9b). The side lobes are important only for low sampling frequency, $f < 0.5$, as otherwise no image energy is present at these spatial frequencies. Although aliasing is somewhat reduced, this simple method compares unfavourably with an ideal band-pass filter (Meer *et al.*, 1987), which transmits all the energy within the pass region and none outside it. We have suggested that some aliasing has a beneficial effect in improving modelling accuracy, and hence it is doubtful in the present circumstances whether use of an "improved" filter would be appropriate, although a preliminary investigation suggests that anti-aliasing may be beneficial (§8.3).

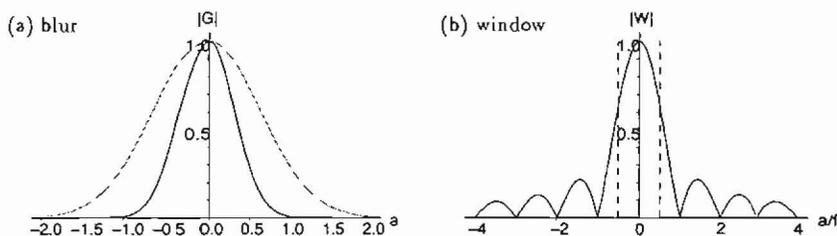


FIGURE A.9: Spectra of blur and window transfer functions. For uncorrected pixel images, aliasing occurs for frequencies a outside the range $-0.5 < a < 0.5$. (a) blur transfer function $|G(b)|$ for $b = 0.25$ (dotted) and $b = 0.5$ (solid). (b) window transfer function $|W|$. The shape of the function is the same for different smooth-sampling frequencies f . An ideal band-pass function only transmits frequencies between the dotted lines.