This is to certify that the

thesis entitled

Markov Random Field
Texture Models

presented by

George Robert Cross

has been accepted towards fulfillment
of the requirements for

Ph. D. degree in Computer Science

Anil Kumar Jain

Major professor

Date 11/12/80

MARKOV RANDOM FIELD TEXTURE MODELS

By

George Robert Cross

A DISSERTATION

Submitted to
Michigan State University
in partial fulfillment of the requirements
for the degree of

DOCTOR OF PHILOSOPHY

Department of Computer Science

1980

ABSTRACT

MARKOV RANDOM FIELD
TEXTURE MODELS

By

George Robert Cross

We consider a texture to be a stochastic, possibly periodic, two-dimensional image field. A texture model is a mathematical procedure capable of producing and describing a textured image. A detailed account of the current literature about texture models is given, along with a discussion of the intuitive notions of texture.

We explore the use of Markov Random Fields as texture models. The binomial model, where each point in the texture has a binomial distribution with parameter θ controlled by its neighbors and "number of tries" equal to the number of gray levels, was taken to be the basic model for the analysis. This represents the first use of the binomial model for image analysis and the first application of the binomial model for any purpose.

A method of generating samples from the binomial model is given followed by a theoretical and practical analysis of its convergence. Examples show how the parameters of the Markov Random Field control the strength and direction of the clustering in the image. The power of the binomial model to produce blurry, sharp, line-like, and blob-like images is demonstrated.

Natural texture samples were digitized and their parameters were estimated under the Markov Random Field model. The use of a hypothesis test for an objective assessment of goodness-of-fit under the Markov Random Field model is a key feature of this investigation. Overall, highly random textures fit the model well.

The estimated parameters of the natural textures were used as input to the generation procedure. The synthetic micro-textures closely resemble their real counterparts, while the regular and inhomogeneous textures do not.

Co-occurrence matrices of binary, first-order Markov Random Field textures were computed as a function of the field parameters. There is a good correspondence between the predicted matrices and the observed matrices on numerous samples.

Finally, a realistic appraisal of the Markov Random Field model as a texture model is accompanied by a list of problems for future study.

To Ellen

ACKNOWLEDGEMENTS

I thank my major professor Anil K. Jain for his help and encouragement during the preparation of this thesis and, more important, for guiding me toward a career in Computer Science. Professor Richard Dubes deserves special recognition for his careful reading of numerous preliminary drafts and suggestions for improvement.

I also thank Professors Page and Mandrekar for serving on my committee and Professor Hedges for welcoming me into the department. Thanks are due in addition to the PR/IP Laboratory graduate students for their assistance during the hectic days of thesis preparation: Neal Wyse, Steve Smith, Wei-Chung Lin, and Gautam Biswas.

The partial support of NSF Grant ECS-8007106 is also gratefully acknowledged.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF SYMBOLS

Symbol	Usage	Section
a	Markov Random Field Parameter for {0,1} variables	3.3
α	Markov Random Field Parameter for {-1,1} variables	5.1
b(i,.)	Markov Random Field ith-order isotropy parameter	3.3
b(i,k)	Markov Random Field ith-order, kth direction parameter	3.3
BB1	Horizontal black-black join count	5.3
BB2	Vertical black-black join count	5.3
BW1	Horizontal black-white join count	5.3
BW2	Vertical black-white join count	5.3
C	Co-occurrence matrix	2.1
F	Clique function	3.2
f(n;i,j)	First-passage probability	4.2
G	Number of gray levels	1.2
γ(1)	Markov Random Field horizontal clustering parameter	5.2
γ(2)	Markov Random Field vertical clustering parameter	5.2
L	Lattice or image	3.2
m(j,j)	Mean recurrence times	4.2

N	Image dimension	1.2
p(i,j)	Transition probability	4.2
p($\underline{X}$)	Coloring probability	3.2
p(X|.)	Conditional probability of X, given its neighbors	2.2.7
{q(j)}	Limiting distribution	4.2
Q($\underline{X}$)	Log probability ratio	3.2
S	Markov Random Field mean	6.2
{S(i)}	States of Markov chain	4.2
S(i,k)	Sum of neighbors at order i, direction k	3.3
T	Weighted neighborhood sum	3.4
θ(T)	Binomial model parameter	3.5.2
U1	Horizontal join count	6.2
U2	Vertical join count	6.2
V(i,j)	Interaction parameter	3.3
WW1	Horizontal white-white join count	6.3
WW2	Vertical white-white join count	6.3
WB1	Horizontal white-black join count	6.3
WB2	Vertical white-black join count	6.3
$\underline{X}$	Coloring	3.2
X(i,j)	Gray level at point (i,j)	2.2.6
Z	Partition function	6.2
$\underline{0}$	Coloring of all zeroes	3.2

CHAPTER 1. INTRODUCTION

Image modelling is a central part of an image understanding system. It is also the core of classical image processing: image restoration [6], and two-dimensional filtering [42]. The subject of image modelling involves the construction of models or procedures for the specification of images. These models serve a dual role in that they can describe images that are observed and also can serve to generate synthetic images from the model parameters. We will be concerned with a specific type of image model, the class of texture models. Understanding texture is an essential part of understanding human vision.

In this thesis, we will explore the use of a Markov Random Field model for the generation and analysis of textured images. The goal of the research is to produce a texture analysis and synthesis system which will take as input a stochastic texture, analyze its parameters according to the Markov Random Field model, and then generate a textured image that both resembles the input texture visually and matches it closely from a statistical point of view. This can be considered a kind of Turing test for image generation [100], in that the proof of the

viability of the system will be to produce textures that cannot be distinguished by humans from their real counterparts. We will not perform a rigorous psychological study of the correspondence, but will concentrate on the statistical evaluation of the goodness of fit of the observed texture and the generated texture.

1.1 Intuitive Notions of Texture

We mention a study by Tamura et al. [99] which attempted to find statistical features corresponding to the usual attributes of texture. Although the study had limited success, the textural attributes identified serve as a useful framework for the discussion that follows. The study delimited six attributes: coarseness, contrast, directionality, line-likeness, regularity, and roughness.

Coarseness refers to the size of the cells (areas of near equal brightness) present in the picture. A fine grain picture has small cells, whereas a coarse picture has large cells. Large areas of equal brightness are significant visual cues.

The contrast of a picture is determined by local variation in gray tone. A low-contrast picture may have all the gray scale values present, but has few transitions from near black to near white in object boundaries. Before processing a picture, we often normalize the contrast and modify its histogram so that an equal number of pixels is present at each gray level. Moreover, the gray scale is then adjusted so that G levels occur at equally spaced intervals from 0 (black) to 255 (white). Such a transformation distorts the natural contrast in the image and negates the value of this feature.

Directionality refers to whether we perceive the trend of the image to be vertical, horizontal, skewed, or non-directional. The bricks picture, Figure 1, is clearly directional, while the gravel image, figure 2, is non-directional.

Line-likeness refers to the presence of lines in the picture. It is contrasted to a blob-like effect, where there are clusters of equal brightness and of circular shape. Some images have a mixture of both. Figure 3 is an image of Venetian blinds and has a definite line-like structure.

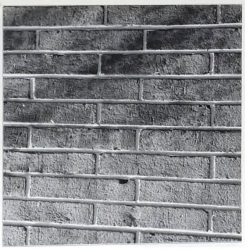


Figure 1. Brick Wall

Figure 2. Coarse Gravel

Figure 3. Venetian Blinds

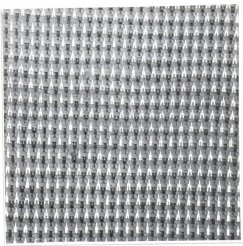


Figure 4. Rag Rug

Regularity refers to whether the image is highly random or not. An image of a rag rug, Figure 4, is clearly regular, whereas the picture of gravel, Figure 2, is irregular.

Finally, roughness describes a tactile attribute of texture rather than a visual one per se. But a rough surface is also visibly rough, and real pictures show surface contours that we would expect to feel rough if we could touch them.

1.2 Texture Formation Paradigms

There is no universally accepted definition for texture. Part of the difficulty in giving a definition of texture is the extremely large number of attributes of texture that we would like to subsume under a definition. Moreover, some of these attributes are apparently contradictory. The first definition is that a texture is a periodic, stochastic gray-scale image. Examples of such textures include cloth, tiles, and bricks. The second definition of texture is any stochastic gray-scale image. This includes such textures as grass, sand, and clouds.

Most texture research can be characterized by the underlying assumptions made about the texture formation process. There are two major assumptions, and the choice of the assumption depends primarily on the type of textures to be considered in the study. The first assumption, which is called the placement rule viewpoint, was explained by Rosenfeld and others [86], [111]. In this model, a texture is composed of small primitives. These primitives may be of varying or deterministic shape, such as circles, hexagons, or even dot patterns. The textured image is formed from the primitives by placement rules which specify how the primitives are oriented, both on the image field and with respect to each other. Examples of such textures include tilings of the plane, cellular structures like tissue samples, or the picture of bricks, Figure 1. Notice that no two bricks are identical, but there is a uniformity to their placement. There is a random aspect to the brick picture in that the shading of the individual bricks varies.

The image of gravel, Figure 2, is not appropriately described by a placement model. The key feature of this image is the coloring and distribution of the cells (areas of near-equal brightness). The primitives are very random in shape and cannot be easily described. The second

viewpoint regarding texture generation processes involves the stochastic assumption. We have already seen that the placement rule paradigm for textures may include a random aspect. In the stochastic point of view, however, we take a more extreme position and consider that the texture is a sample from a probability distribution on the image space. The image space is usually an N by N grid and the value at each grid point is a random variable in the range $\{0,1,\ldots,G-1\}$.

1.3 Vision Research about Texture

In this section, we will describe what vision researchers have found to be the important constituents of human texture vision. Although machine perception need not be an emulator of the human visual system, the success of human vision in discriminating textures merits attention. In addition, since we are interested in performing automatically tasks that are currently done by humans, such as radiographic screening, we should be guided by the cues and sensitivities of the human visual system.

1.3.1 Julesz' Work on Texture

Julesz' investigations of texture span twenty years. Besides the study of texture perception itself, he has studied texture's role in binocular or stereoptic perception, motion perception, and color perception [57]. We will examine only the work on texture itself.

Early in his work, Julesz decided that it is appropriate to examine perception of textures without visual cues. He uses random dot patterns controlled by analogs of one-dimensional Markov processes and Gaussian processes. Because of this, his work has limited applicability to textures generated according to placement rule schemes. His work is also concentrated on "pure perception," which means perception in the absence of higher-order scrutiny. When he speaks of "effortless discrimination," he refers to the process of deciding whether two textures are the same after seeing them for only 100 milliseconds. If they are seen for a longer time than that, higher-order conceptual processes take over.

One of the key features that he identified in texture perception is cluster formation [56]. This is related to image coarseness in the sense mentioned earlier in section

1.1. He found that the human visual system acts as a slicer or thresholding device in clustering intensity values. For example, if the image contains the gray levels 0, 1, 2, and 3, then it is not possible for clusters to form consisting of the gray level sets {0,2} and {1,3}. Further, cluster detection is central to the entire perceptual process, from the lowest level aggregation up to the higher order mental processes.

The aspect of Julesz' work that has captured the most attention is the conjecture that the human visual system cannot effortlessly discriminate textures that agree in their second-order statistics [54], [55], [58], [59], [60]. The first-order statistics are simply the proportions of the pixels at each gray level. The proportions control whether we see the picture as a black object on a white background or vice versa. Julesz has found that we can effortlessly perceive a first-order difference between textures within fairly narrow ranges [54]; this work is related to the classical figure-ground problem and numerous optical illusions [35]. The second-order statistics involve the joint probability that a randomly thrown rod of length r will fall on the image in such a way that one of its endpoints lands on a pixel of color i while the other lands on a pixel of color j.

Similarily, third-order statistics have to do with triangles and the configuration of gray levels at the vertices after a similar random experiment. The conjecture that the second-order description (which of course implies the first-order statistics) determines our discrimination ability has held up remarkably well with only a few classes of unusual textures that confound it [58].

Another aspect of texture that Julesz investigated [57] is that periodicity of a texture is only observable if the frequency is large compared to the size of the picture. For example, it is all but impossible to notice that Figure 5a consists of the same texture repeated four times, whereas it is immediately obvious that Figure 5b consists of repetitions of the same random structure. The difference is that Figure 5b consists of sixteen repetitions of a 16 by 16 picture in a frame size of 128 by 128, whereas Figure 5a consists of only four repetitions of a 64 by 64 picture in the same frame size of 128 by 128.

Julesz' work has been extended by a number of researchers, notably Pratt et al. [83]. It is now generally assumed that the second-order statistics are

Figure 5. Repetitions of Random Textures: (a) Four repetitions (b) Sixteen repetitions.

sufficient in a pattern recognition environment to discriminate between textures. There is still much work to be done in the sense that the second-order probabilities must be combined in some way to provide features for recognition of the textures. The previously discussed work of Tamura et al. [98] is such an attempt, as is the continuing efforts of Conners and Harlow [23] to define a texture recognition feature set from the second-order statistics. It is possible to use third-order statistics in texture discrimination even though the human visual system does not appear to make use of them. The difficulty lies in attempting to estimate

the third-order parameters given a relatively small texture field, such as a field of size 64 by 64. Moreover, the storage requirements for maintaining frequency statistics on the large number of possible configurations are enormous.

1.3.2 Marr's Work on Texture Vision

The ultimate goal of Marr's research is a comprehensive computational theory of vision. Such a theory can be implemented with a computer, which will, in fact, emulate the human visual system. His plan revolves around the "primal sketch" [68], [69]. The primal sketch is produced from the original intensity values received at the retina.

The primal sketch consists of measurements on the strengths and characteristics of the following primitives: edges, lines or bars, and blobs. These primitives are quantified further by the predicates of orientation, size (length, width, diameter, or aspect ratio if appropriate), contrast, position, and terminal points. This is in general agreement with the work of Caelli and Julesz [18], [19] in describing the processing at early levels in the visual system.

Marr asserts that once the primal sketch is computed, it can then be analyzed to find objects or regions of interest. This is done by aggregating the primitives which are close in some sense. We may thus characterize his overall approach as "bottom-up," in the sense that we do not begin with high-level knowledge of the scene or preconceptions of the objects present. The recognition of objects using the primal sketch model is based on using the texture information to define the objects, rather than considering the texture to be noisy background.

1.4 Applications of Texture

There are essentially four areas of image processing in which texture plays an important role. We offer this discussion as justification of our interest in the production of synthetic textures. As a side benefit, such generation procedures may help in furthering our understanding of human perception. According to Julesz [57],

> ... to solve the problem of visual texture generation of familiar surfaces is important for both theoretical and practical uses. I can only hope that in the near future scientists will learn to clarify the enigmatic problem of familiar textures.

(i) Classification

Suppose, for example, that we have a series of pictures of mineral samples that we would like to classify into groups based on the type of mineral represented. The true class of some training samples is generally known and could be provided by geologists. Such images have a special structure characteristic of the mineral, but the structure is not deterministic like a tiling. We would like to specify some features of the image in order to classify it. There are obvious differences in distribution of grain size, gray tone, and even color. Studies of this type have been done by Haralick [46] on sandstone samples and by Weszka et al. [106] on terrain samples.

(ii) Image Segmentation

Segmenting an image means dividing it into homogeneous regions such that each region has some significance. Typically, the regions are connected and we have some a priori idea of what we are looking for or what might be present in the image. For example, we may wish to segment a LANDSAT image into land-use categories by means of texture analysis. Nevatia [76] uses the

assumption that the objects of interest have strong boundaries and the textured background is composed of short line segments to extract the boundaries of a small toy tank from the background of grass. Thompson [99] uses a generalized edge operator to define texture borders and is successful in finding the boundaries between two regions with differing textures. A linear approach to segmentation of textured regions is taken by Deguchi and Morishita [26], which we will discuss in detail in chapter 2.

(iii) Realism in Computer Graphics

Computer graphics differs from image processing in that its goal is the production of images from a description, whereas image processing is concerned with the modification and interpretation of real-world images. Although high resolution, color, raster-graphics displays capable of producing realistic images are available, the existing algorithms cannot produce the desired realism.

Texture gives important information on the depth and orientation of an object, besides being an intrinsic feature of realistic objects [64], [78]. A number of approaches have been taken, mostly related to mapping a

flat textured image onto the skeleton of a three-dimensional object and then displaying a perspective view. Catmull [20], [21] describes a procedure for wrapping photographs of texture onto objects to produce textured surfaces. Blinn [14], [15] uses a patch of texture to tesselate the image. Csuri et al. [25] extend the basic patch idea to include the specification of structure for the patch in terms of reflectance values at each pixel. This allows more realistic shading and permits the simulation of the appearance of the textured patch under the generated illumination. They intend to extend their model to include some randomness in the patches. Dungan [29] has coupled height data from Defense Mapping Agency maps with an overlay identifying terrain segments (trees, lakes, etc) to simulate the appearance of the ground scene to a viewer in an airplane.

As the applications of computer graphics grow, the importance of an understanding of texture will also increase. In fact, the generation of natural textures will likely play a more important role than the recognition of textures.

(iv) Picture Encoding

The final area where texture generation and analysis has application is in picture encoding. If we can generate a texture with a few parameters that is indistinguishable from an observed texture, then we have effectively compressed the enormous amount of data in the original texture to the parameter vector. By analogy, this is tantamount to knowing an algorithm for the generation of normal deviates rather than carrying an enormous pre-printed book listing millions of samples from a normal population.

Such a procedure has application to the previously mentioned graphics problems in that we could artificially generate background scenes from descriptions defined by textural parameters. Such scenes are presently generated by a hybrid graphics and image processing procedure of mapping photographs onto the graphics skeletons.

Some work has already used texture synthesis in the pure information theory context of reducing the total number of bits needed to store the picture. Modestino and Fries [75] encode a picture according to a Poisson line model, while Delp et al. [27] use a Gaussian model. In

both cases, the image is partitioned into small squares. The texture of each square is estimated and the few parameters of the texture are stored instead of the 256 gray levels of the observed texture. Since the picture is reconstructable by the same model that was used to estimate it, the picture can be approximately reconstructed square by square from the estimated parameters.

1.5 Organization of the Thesis

Chapter 2 contains a review of texture models that have appeared in the literature and further specifies the levels of texture modelling. Chapter 3 describes the specifics of the Markov Random Field theory needed to understand the models that we use in generating and fitting textures. Chapter 4 details the generation procedure for obtaining a texture sample from a distribution specified by a parameter set. In chapter 5, we give our results in fitting a variety of models to a sample of textures from the Brodatz texture album [17]. Chapter 6 gives theoretical results in deriving textural features from the model parameters, such as the co-occurrence matrices for Markov Random Field textures. Finally, chapter 7 gives our conclusions and directions for future research.

CHAPTER 2. TEXTURE MODELS

2.1 Introduction

By a model of a texture, we mean a mathematical process which creates or describes the textured image. The goal of texture modelling is the description of the image; real textures can be compared to generated textures as a test of the validity or utility of the model. The test can take the form of a psychological study or of a statistical assessment of goodness of fit.

A secondary goal of texture modelling is classification of textures. The numerical parameters of the model can be used as features to classify the texture. There is a distinction between model-based studies and attempts to find good features for the classification of textures. In a model-based environment, we have the capability to produce, for example, textures that match observed textures. In a feature-based texture analysis, the textural features are measured without an ideal or representative texture in mind.

Texture modelling can be approached at three levels. The highest level, which we call the knowledge representation level, uses our understanding of the physical scene in order to describe the textured image. For example, consider the set of textured images of wood grain observed in samples {T(θ)}, where θ is the angle at which the cut was made in the tree to form the wood sample. Because trees grow in concentric rings, the ideal appearance of such samples T(θ) varies continuously from an image of concentric circles (when θ is 0) to ellipses, and finally to parallel lines (when θ is 90 degrees). This ideal image is, of course, corrupted by noise, degraded by irregularities in the circles, and, perhaps, distorted from the expected image by the presence of knots. We do have, however, an ideal image for specifying membership in the class defined by θ. Such a model was used in automatic drafting [108].

Once we have this ideal model, the classification problem can be attacked. In the present example, the classification problem has been reduced to the relatively trivial one of circle and line detection along with estimation of the aspect ratio of the ellipses. Thus, our structural description of the expected image permits us to identify observable features. On the other hand, the

knowledge used in one domain of images may not be applied easily to other areas. Models constructed for the texture of corn fields from satellite photographs may not be of much use in constructing a model for the appearance of wheat fields or even corn fields grown in other countries under different cultivation regimes. The success of the knowledge representation approach is limited by our capability to generalize.

The bottom level of texture modelling is called the feature-based level. In fact, it is possible to make a case for not calling feature-based studies modelling at all. We assume that the textures are samples from some unknown probability distribution over a lattice or grid L, representing the image, which has dimension N by N. The objective is to find features that correctly classify a texture into one of the texture classes represented by a set of different probability distributions. A survey of the well-known and commonly used texture features that have been investigated appears in a recent report [47].

Texture features include the image autocorrelation function [62], which has been found to have visual significance. An influential paper by Haralick [45] describes the use of features computed from the gray level

co-occurrence matrices. The (r,s) entry in a co-occurrence matrix is the probability that a point of the image (i,j) has gray value r while point (k,l) has gray value s, where k = i + dx, and l = j + dy. The increments dx and dy are constant over the individual matrices and we compute a set of matrices indexed by the increments (dx,dy). If G is the number of distinct gray levels in the image, then the co-occurrence matrices are of size G by G.

Haralick [45] defines some 14 features based on the co-occurrence matrices. We mention three of them. Let R be the number of pairs of grid positions that were used to compute the co-occurrence matrix C with entries c(i,j) corresponding to the displacement (dx,dy). The angular second moment feature f1 is defined by

$$f1 = \sum_{i=0}^{G-1} \sum_{j=0}^{G-1} \left(\frac{c(i,j)}{r} \right)^2$$

and measures the homogeneity of the image. Large values of f1 indicate the presence of many dominant gray level transitions in the image. The contrast feature is given

by f2

$$f2 = \sum_{n=0}^{G-1} n^2 \sum_{|i-j|=n} \left(\frac{c(i,j)}{r} \right)^2$$

and is a measure of the local correlation. Small values of f2 correspond to images with little local variation. Finally, the correlation feature f3 measures the linear dependence in the image:

$$f3 = \sum_{i=0}^{G-1} \sum_{j=0}^{G-1} \frac{(ijc(i,j)/R) - UxUy}{SxSy}$$

where Ux and Uy are the means and Sx and Sy are the standard deviations of the marginal distributions associated with c(i,j)/R. The feature f3 is large if the correlation in the direction expressed by (dx,dy) is large.

Several studies have shown that features based on co-occurrence matrices perform reasonably well in classifying some textures, [106], [23]. Conners and Harlow [24] have theoretically evaluated co-occurrence features in distinguishing Markov scan textures. Other features that have been used with some success include gray level run lengths [37], edge per unit area [86], [87], extrema in gray scale height [86], and encoded maxima along scan lines [74].

2.2 Stochastic Texture Modelling

We now turn our attention to the middle level of texture modelling, the stochastic process approach. We use the term stochastic process in a loose sense to describe a random procedure used to generate an image. There is some inevitable overlap between the stochastic process approach and the knowledge representation point of view since we try to find a stochastic process for the modelling of the image that is physically meaningful and related to the textures which we are modelling. We use more prior information about the texture in the stochastic process approach than in the feature-based approach. In the stochastic process approach, brightness levels or pixel gray values are the random variables. The level

X(i,j) at some point (i,j) is not independent of the levels at other points in the image. In fact our principal concern is about the correlations between the {X(i,j)}.

This section is subdivided into discussions about the major classes of texture models that have appeared in the literature. A short description of each type is given, along with some indication of the class of textures that can be handled by the model.

2.2.1 Quantized Continuous Models

Suppose that we have M points in the unit square. We impose an N by N grid on the unit square and then count the number of points that fall in each grid square. Such a grid can now be quantized by a scale change to form an image in the usual sense, with a gray scale in the range from 0 to G-1. The value of G can be either the maximum number of points that fall into a grid square or we may choose G *a priori* and then use a quantizing function to map the accumulated grid counts to levels from 0 to G-1.

A wide variety of continuous processes, including the multivariate normal distribution, are available to form such images. On the other hand, much greater generality can be achieved by the use of spatial processes as explained by Ripley [84] or Matern [70].

Following Ripley [84], a realization of a spatial process is a countable set of points without limit points. For each measurable set A, let Z(A) be the number of points of a realization that fall in the set A. If A is a bounded set, then Z(A) is finite. The process is described by the set of random variables {Z(A)| A is a bounded Borel set}. If the process is second order, translation invariant, and rotation invariant, then Ripley calls the process a 'model'. These latter assumptions are for mathematical tractability. Ripley remarks that the first and second moments of a spatial model should be sufficient to distinguish realizations and justifies this by the work of Julesz [54], who found that, for the most part, textures with the same second-order statistics were indistinguishable by human vision (see chapter 1).

The continuous models of Ripley [84] can be broadly delimited on a scale of clustered, random, and inhibitory. Let U be a set of unit area. The intensity of the process

is called λ and we assume that $E[Z(U)] = \lambda$. For a given λ, we would expect a concentration of nearest neighbor distances for a clustered process at levels far below that for a random (i.e. Poisson) process. Similarly, an inhibitory model has a concentration of nearest neighbor distances far above that expected for the random process. One kind of inhibitory model is called hard core, in that we do not allow any points less than a distance $2r$ apart, where r is a positive parameter of the process called the inhibition distance.

The discretization of continuous processes as image models of texture has not been explored except for a brief mention in [90] and some simulation in [11]. Some results in discrete analogs of continuous processes appear in Rogers' monograph on retail trade [85], but no direct application to images was made.

2.2.2 Time Series Models

McCormick and Jayaramamurthy [66] have developed a model of texture based on a time series forecasting technique. A time series is, of course, a one-dimensional sequence of random variables indexed by a parameter which is usually associated with time. To make such a sequence

into an N by N image, the time series is broken up into pieces of size N and then stacked as rows of an image.

McCormick and Jayaramamurthy use the auto-regressive integrated moving average process (ARIMA) to simulate textures. The parameters are estimated from real textures. They exhibited some success in generating textures characterized by long streaky lines such as the cheesecloth picture in Brodatz [17], image D105.

The applicability of this model to isotropic textures, i.e., ones with random or irregular globular clusters, is dubious. It is, however, an appropriate model for line-like textures, as they are able to satisfy the criterion expressed in Chapter 1 that the texture synthesis process be successful in the production of an image that the viewer might think was a real texture.

2.2.3 Fractal Textures

The term fractal was coined by Mandelbrot [67] to describe point sets or stochastic processes whose Hausdorff dimension exceeds their topological dimension. The topological dimension of a point set usually corresponds to our intuitive notion of dimension but

differs for certain pathological cases. As we should expect, the topological dimension of a countable set of points is 0, of a line is 1, and of a plane is 2. A discussion of the definition and the development of the theory of topological dimension is given by Hurewicz and Wallman [51]; a more modern treatment with considerable detail on special cases of dimension in abstract spaces is given by Pears [81]. The Hausdorff dimension of a set is always at least as large as its topological dimension. Sets that are extremely irregular, such as nowhere-differentiable curves and surfaces, have Hausdorff dimensions which exceed their topological dimensions. For example, the Cantor set has Hausdorff dimension equal to log2/log3, though its topological dimension is 0. Moreover, the Hausdorff dimension of a set in Euclidean D-space can be a fractional value but is always less than or equal to D [51].

A fractal set process can generate a texture in a number of ways. Mandelbrot [67] exhibits some Brownian textures. These are simulations of mountain ranges Z=f(X,Y) over the X-Y plane such that every plane cut in a direction perpendicular to the X-Y plane gives an ordinary planar Brownian function. The displayed image is produced by simulating a light source at an angle of 60 degrees

with the resultant shadow effects, and viewing angle of 25 degrees. The pictures look like irregular surfaces, such as lunar lanscapes.

Other textured images can be obtained by dropping on a plane circles whose radii vary according to a random variable with a Pareto distribution, with the exponent in the density function playing the role of the Hausdorff dimension. Work is needed in the estimation of the fit between natural textures and fractal textures before fractal models can be utilized as true texture models rather than just pattern generation processes. In addition, further work is needed on the estimation of the Hausdorff dimension of stochastic processes. For the appropriate background see Billingsley [12], [13].

2.2.4 Random Mosaic Models

A tiling or tesselation of the plane is a collection {A} of disjoint sets whose union is the entire plane. The elements of {A} are called cells. A mosaic is a tesselation combined with a function H from {A} to the finite set of gray levels {0,1,...,G-1} which is constant on each cell: H is called the coloring function.

We may speak of random tesselations where the tesselations are determined by some random process or where the H function is probabilistic, for example controlled by a discrete Markov process. We generally only deal with mosaics on bounded sets; such mosaics can be achieved by taking a full planar mosaic and intersecting C with the bounded set and restricting the coloring function. Such a restriction is the true environment for the image processing applications, yet it is usually easier to deal with the theory in the infinite plane case and hope that the bounded set lattice is large enough to allow reasonable approximation.

In a sense, the mosaic models generalize the standard notion of a digitized image. The pixels of an image correspond to the cells of a mosaic, but the mosaic models allow us to consider cells that are much larger and of different shape. Ahuja [2] makes a distinction between region-based models and pixel-based models. The appeal of the region-based models is that they have the virtues of the stochastic approach along with some of the strength of the placement paradigm in describing large primitives. For example, the bricks picture, Figure 1, is an example of a macro-texture appropriately described by a region-based model. The relatively fine-grained gravel

picture, Figure 2, is a micro-texture, best suited to a pixel-based model.

The cells can be generated in a number of ways. We follow current literature [75], [89], [90] and discuss the possible patterns.

(i) Random Checkerboard: A checkerboard is oriented at an angle θ to the x-axis.

(ii) Occupancy Model: Use any spatial point process to generate a realization in the plane. Define the cells V(p), indexed by the points p of the realization by the relation: $V(p) = \{q \text{ on the plane} \mid d(q,p) \text{ minimal among } p\}$. Each point of the plane is a member of exactly one V(p) since the realizations have no limit points. This model is also called the Voronoi polygon model.

(iii) Poisson Lines: Distribute lines isotropically over the plane using a Poisson process to generate the radii (distance from line to origin) and the angle the line makes with the x-axis. The polygons determined by the lines are convex and constitute the cells of the mosaic.

(iv) Bombing Patterns: Let B be a convex set (this restriction is not necessary but makes the process mathematically tractable). The set B is placed on the plane with random orientation at a point x which is determined by a "marked" Poisson process. Marking a process means that we enumerate each point in a realization and assign it a positive integer. We assume that the copies of the bomb B are of one color, say black, while the plane is white. The cells of the mosaic are then the components of the bombing process. After the bombing is over, the plane can be recolored using any coloring function for the components.

(v) Johnson-Mehl: Points are dropped on the plane using a marked process and the cells are formed by growth from these "seed" points. Unlike the occupancy model, the resulting cells need not be convex. This model is appropriate as a physical model for crystal growth in metallurgical applications, but has proven to be mathematically intractable for expressing auto-correlations and other features.

(vi) Dead Leaves: The dead leaves model is similar to the bombing model except that the boundary of the bombing shape is retained and the boundaries of previously dropped

"leaves" are removed. The final pattern resembles the appearance of leaves on the forest floor with many full outlines of the shapes visible and many partial shapes. This model is discussed by Serra [93].

The mosaic models discussed above can be subsumed into a more general theory of random sets called Mathematical Morphology. The theory is under continuing development by a number of European workers and has seen considerable application in the metallurgical area. See the work of Serra [92], Matheron [71], [72], and Giger [40]. Serra calls this general class of models Boolean models. The Boolean model starts with a realization of a Poisson process and takes each point to be a growth center. The primary grain is a random set, usually convex, the randomness being controlled by a growth process. It is clear that this very general growth process can include the above mosaic models. Algebraic operations, including the familiar operations of union, intersection and specialized operations called dilation and erosion, allow the specification of composite models. Special hardware is available to perform the set operations along with the estimation of model parameters [63], [91], [95], [96].

The advantage of some of the mosaic models is that the theoretical autocorrelations can be computed from a knowledge of the process parameters. By fitting the observed autocorrelation to the theoretical autocorrelation for the proposed model, features such as cell size distribution can be inferred without having to segment the image. Julesz [54] has found that the distribution of plateau regions of constant gray level is a very important consideration in texture discrimination, so any model that effectively models cell size contributes to our understanding of texture.

2.2.5 Syntactic Methods

The syntactic approach to texture, like the stochastic methods, can be dichotomized into modelling and classification efforts. Ehrich and Foith [30], bolstered by the success of syntactic methods in waveform analysis, have used syntactic methods to encode textures as mountain ranges whose heights are associated with gray levels. The encoded texture can then be used in classification experiments.

Our notion of a model includes the capability to synthesize samples from a textural class. Lu and Fu [65] take this approach and encode 9 by 9 squares of a texture as a primitive in a stochastic grammar and then link the squares together to form an image. Some success is reported in generating Brodatz [17] textures that have a fairly regular structure, such as the picture of reptile skin. We suspect that difficulty would be experienced in using this method to synthesize micro-textures such as sand or grass. Other problems include the extension to multiple gray levels, but the principal problem is the estimation (or in this context, grammatical inference) of the model parameters.

2.2.6 Linear Models

Following a procedure analagous to a linear filtering model, Deguchi and Morishita [26] assume a texture model of the form:

$$X(i,j) = \sum_{(p,q)\ (\text{not } (0,0))} A(p,q)X(i-p,j-q) + E(i,j)$$

where $X(i,j)$ is the gray level at point (i,j), and the

E(i,j) are i.i.d Gaussian variables with zero mean. The local neighborhood is a rectangle of dimension 2m+1 by 2n+1.

The A(p,q) parameters are estimated by minimizing the expected difference between the estimated X(i,j) and the observed X(i,j). This approach is fairly similar to the classical analysis of Whittle [107], but Whittle was only interested in estimating the parameters in an applied statistics context for testing the hypothesis of no interaction between the site variables. When we begin an estimation problem, we do not know what value of m and n to choose. Hence, we must consider them parameters of the model which must be estimated. Deguchi and Morishita use a procedure due to Akaike [4], [5] in a time series context to estimate m and n in an optimal manner.

Textures following the above model can be generated by a filtering scheme. Generate independent Gaussian variables at each site. Then use the filter coefficients A(p,q) to create a new image by summing over the neighborhoods (it is, of course, possible to use an FFT algorithm to do this faster). The real values at each point of the lattice can then be quantized.

Deguchi and Morishita do not generate any textures from the above model. Their purpose is to segment the textured regions of an image. One example is the location of small regions, called "defects", whose texture differs from that of a background texture. To locate the defects, first estimate the A(p,q) parameters for the entire image. Since the defect regions are presumed small, they will not significantly influence the estimation of the A(p,q) parameters. The error between the predicted pixel values using linear estimation and the true value is calculated for each pixel in a 2m by 2n neighborhood. If the error exceeds a preset threshold, then the pixel is presumed to belong to the defect region. Variations on this theme are presented for other segmentation experiments.

2.2.7 Markov Models

The term "Markov" is loosely applied to any model where the probability of any particular gray level at a site (i,j) does not depend on pixels beyond a "small" neighborhood of (i,j). We may express this by the relation:

p(X(i,j)=k|given all other sites) =

p(X(i,j)=k|values at neighbors).

The term "neighbors" is defined topologically and can be enforced as any configuration surrounding a given pixel. Following Besag [10], or Hammersley [44], a Markov Random Field obeys the above neighborhood condition along with two other conditions. The first is called positivity and says that any assignment of colors to the lattice (integers in the range {0,1,...,G-1}) has non-zero probability. The other condition is called homogeneity and says that the conditional probabilities are unaffected by the actual coordinates of the site; only the neighborhood values matter. The Markov Random Field model has been briefly investigated by Hassner and Sklansky [48], [49], [50]. Their work was limited to an exposition of the equivalence between the Gibbs field and Markov Random Field expressions for the conditional probability distributions (see Spitzer [94] for the proof) and generation of a few examples of textures. Moreover, they limited their attention to the binary case.

An alternative to the use of the full two-dimensional Markov Random Field for images is to traverse the lattice along a scan line and provide a direct analog to the usual Markov chain [1]. Connors and Harlow [24] generated textures according to a simple Markov chain on the rows, which produces streaky line textures but ignores the correlations between pixels in neighboring rows. Haralick and Yokoyama [110] generated essentially one-dimensional textures using scans, but provided some correlations between neighboring rows by considering changes in the features computed from the co-occurrence matrices.

2.3 Summary

We have defined a texture model to be a process capable of generating or describing a texture. The types of processes capable of modelling texture were delimited as knowledge-based descriptions, stochastic processes, and feature-based descriptions. We concentrated on the middle level stochastic process models and surveyed the models available. These include discretized continuous models, time-series, fractals, random mosaics, syntactic methods, linear models, and Markov models. Each model was found to be useful for certain classes of textures. It is as inappropriate to assume that one class of generation

procedures can generate all textures as it is to assume that the Gaussian distribution can explain all data.

CHAPTER 3. MARKOV RANDOM FIELDS

3.1 Introduction

The brightness level at a point in an image is highly dependent on the brightness levels of neighboring points unless the image is simply random noise. In this chapter, we explain a precise model of this dependence, called the Markov Random Field. The notion of near-neighbor dependence is all-pervasive in image processing. Focusing directly on this property is a promising approach to the overall problem of micro-textures. The Markov Random Field has had a long history, beginning with Ising's 1925 thesis [53] on ferromagnetism. Although it did not prove to be a realistic model for magnetic domains, it is approximately correct for phase-separated alloys, idealized gases, and some crystals [77]. The model has traditionally been applied to the case of either Gaussian or binary variables on a lattice. Besag's paper [10] allows a natural extension to the case of variables that have integer ranges, either bounded or unbounded. These extensions, coupled with estimation procedures, permit the application of the Markov Random Field to image and texture analysis.

3.2 Definitions and Theorems

Our exposition follows Besag [10] and Bartlett [7]. As before, let X(i,j) denote the brightness level at a point (i,j) on the N by N lattice L. We simplify the labelling of the X(i,j) to be X(i), i = 1,2,...,M where M = N**2.

Definition 3.1: Let L be a lattice. A coloring of L (or a coloring of L with G levels) denoted $\underline{X}$ is a function from the points of L to the set {0,1,...,G-1}. The notation $\underline{0}$ denotes the function that assigns each point of the lattice to 0.

Before defining a special type of probability on the set of all $\underline{X}$, we first give the notion of neighbor. Note that the definition does not imply that the neighbors of a point are necessarily close in terms of distance.

Definition 3.2: The point j is said to be a neighbor of the point i if

p(X(i)|X(1),X(2),...,X(i-1),X(i+1),...X(M))

depends on X(j).

Now we can give the definition of a Markov Random Field.

Definition 3.3: A Markov Random Field is a joint probability density on the set of all possible colorings $\underline{X}$ of the lattice L subject to the following conditions:

1.Positivity: $p(\underline{X}) > 0$ for all $\underline{X}$.

2.Markovianity:

$$p(X(i) \mid \text{All points in lattice except } i) = p(X(i) \mid \text{Neighbors of } i)$$

3.Homogeneity: $p(X(i) \mid \text{Neighbors of } i)$ depends only on the configuration of neighbors and is translation invariant (with respect to translates with the same neighborhood configuration).

We would like to delimit insofar as possible the kind of probability distributions that represent Markov Random Fields. This is the content of the Hammersley-Clifford theorem. We first need a definition.

Definition 3.4: A clique is a set of points that consists of either a single point or has the property that each point in the set is a neighbor of every other point in the set.

It will prove to be convenient to consider the distribution function in terms of the ratio of $p(\underline{X})$ to $p(\underline{0})$. Define the quantity $Q(\underline{X})$ by:

$$(3.1)\quad Q(\underline{X}) = \ln(p(\underline{X})) - \ln(p(\underline{0})).$$

We may expand $Q(\underline{X})$ as follows:

$$(3.2)\quad Q(\underline{X}) = \sum_{i} X(i)F_i(X(i)) + \sum_{i,j} X(i)X(j)F_{i,j}(X(i),X(j)) + \cdots + X(1)\cdots X(M)F_{12\cdots M}(X(1),\ldots,X(M)).$$

Each summation in equation (3.2) extends over sets of indices without repetitions.

Note that Q($\underline{X}$) can always be expanded in this way as long as the positivity condition is satisfied. The F functions can be determined inductively from a knowledge of the function P($\underline{X}$). First set every X(j) to zero except for a single index i. This determines the value of Fi. After repeating this for every index i, the F functions with two arguments can be found by setting all pairs {X(i),X(j)} to zero except for a single pair {X(r),X(s)}. Continuing in this manner with successively larger sets of indices yields the values of the F functions for all numbers of arguments. With this notation, we may now state the Hammersley-Clifford theorem.

Theorem 3.1: Q($\underline{X}$), expanded as in equation (3.2), defines a valid probability distribution for a Markov Random Field provided that the F functions with subscripts ij...s are non-zero if and only if the points i,j,...,s form a clique.

proof: See Besag [10], page 196.

The Hammersley-Clifford theorem provides a connection between the purely graph-theoretic relationships on a lattice with the algebraic form of the expansion of the distribution function in equation (3.2). It also assures

that the conditional probability distribution determines the joint probability mass function. In fact, the theorem can be applied to the case where the lattice is triangular, or even irregular. Such extensions allow the formulation of models in geography; for example, see the review paper by Cliff and Ord [22].

3.3 Further Assumptions

In order to simplify the form of the probability density in equation (3.2) further, we make two assumptions:

Assumption 1: The F functions in (3.2) with more than two arguments vanish identically.

Assumption 2: The conditional probability distribution at each point of the lattice belongs to the exponential family.

Definition 3.5: A Markov Random Field is called an Auto-Model if it satisfies assumptions 1 and 2 above.

The class of auto-models is defined in order to simplify the inference problem for the F functions. Under these assumptions, we may write Q($\underline{X}$):

(3.3) $$Q(\underline{X}) = \sum_{i} X(i)F_i(X(i)) + \sum_{i,j} X(i)X(j)F_{i,j}(X(i),X(j)).$$

This results, following Besag [10], in the following form for the conditional densities:

(3.4) $$p(X(i)=k|\ldots)/p(X(i)=0|\ldots) = \exp(kT(\underline{X}))$$

where

(3.5) $$T(\underline{X}) = F_i(X(i)) + \sum_{j=1}^{M} V(i,j)X(j).$$

The {V(i,j)} are the parameters of the model, V(i,j) = V(j,i), and V(i,j) = 0 if and only if i and j are not neighbors of each other.

3.4 Order Dependence

Note that in equation (3.4), we are actually assuming that the parameters V(i,j) may vary with i. In most cases, we are interested in models whose parameters {V(i,j)} depend only on the distance from point i to point j. This prompts the definition of order.

Definition 3.6: On a lattice, the distances between points assume discrete values. The first few terms of the sequence of possible squared distances are 0, 1, 2, 4, 5, 8, 9, 10. See Figure 6 for a picture of this. Call this sequence {e(k)}, k = 0, 1, 2,... Let r = max(d(i,j)) where i, j are points for which V(i,j) is non-zero. A process is said to have order k if e(k) = r**2. Figure 7 shows the order of the points identified in Figure 6.

			10	9	10		
		8	5	4	5	8	
10	5	2	1	2	5	10	
9	4	1	X	1	4	9	
10	5	2	1	2	5	10	
		8	5	4	5	8	
			10	9	10		

Figure 6 Squared distances to the point X

The number of parameters needed to describe a model depends on its order. For example, a first-order model is specified by three parameters. In the case of a lattice with range set {0,1} the conditional probabilities take the form (the points are labelled as in Figure 8):

(3.6) $p(X=x|u,u',t,t') = \exp(xT)/(1 + \exp(T))$

where

(3.7) $T = a + b(1)(t + t') + b(2)(u + u')$.

The binary Markov Random Field described by equation (3.6) has parameters a, b(1), and b(2). The parameters b(1) and b(2) control the clustering in the lattice; positve values indicate attraction between points with value 1, negative values indicate repulsion, while a value of zero implies a random configuration. In the random case, the proportion of black points (1-valued points) should be $\exp(a)/(1 + \exp(a))$. The value of b(1) controls clustering in the E-W direction while b(2) controls N-S clustering.

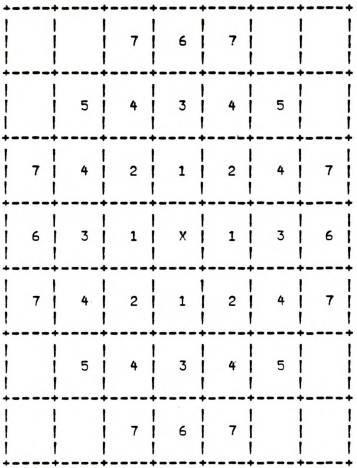


Figure 7 Orders relative to the point X

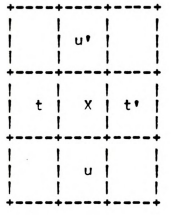


Figure 8 First-order Neighbors

A full second-order model involves cliques of size three and four. The expressions (3.6) and (3.7) are replaced by the following expressions (the points labelled as in Figure 9):

(3.8) $p(X=x|t,t',u,u',v,v',w,w') = \exp(xT)/(1 + \exp(T))$

where

(3.9) $$\begin{aligned} T = \; & a + b(1)(t + t') + b(2)(u + u') \\ & + a(1)(v + v') + a(2)(w + w') \\ & + z(1)(tu + u'w + w't') \\ & + z(2)(tv + v'u' + ut') \\ & + z(3)(tw + w'u + u't') \\ & + z(4)(tu' + uv + v't') \\ & + q(tuv + t'u'v' + tu'w + t'u'w'). \end{aligned}$$

Cliques containing three points are triangles formed by selecting three points of a two by two square. Such a triangle contributes to the conditional probability of X

only when X is one of its vertices. By the homogeneity assumption, each triangle with the same orientation makes the same contribution to the sum T in equation (3.9), and is associated with a certain z(i). For example, following Figure 9, the triangles xtw, xu't', xw'u all have the same orientation and influence the conditional probability in equations (3.8) and (3.9) according to the value of the parameter z(3). For cliques of size 4, we consider two by two squares with one corner on X.

In this thesis, we have considered only the linear case of the full expansion, which means that the F functions with two or more arguments vanish. In the expression for the full second-order model given by

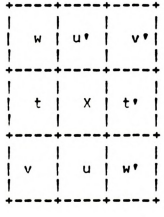


Figure 9 Second-order neighbors

equations (3.8) and (3.9), this implies that the z and q parameters are zero. We have also limited our attention to the case of a maximum fourth-order dependence, since it does not seem likely that any higher order parameters can be estimated accurately from textured images of size 64 by 64.

When dealing with a finite rectangular lattice and using the definitions of neighbor and order above, points on the edge of the lattice have fewer neighbors than the interior points. We compensate for this by assuming that the lattice has a periodic or torus structure. This means that the left edge is connected to the right edge and the upper edge is connected to the lower edge.

Our final form for the value of T is given below for orders up to four. Models of order less than four can be obtained by assuming that the b(i,k), the parameters of the model, are 0 for all i larger than the order.

$$(3.10)\quad T = a + \sum_{k=1}^{2} \sum_{i=1}^{4} b(i,k)S(i,k).$$

The S(i,k) are composed of the sum of the values at order i with respect to X. The direction is defined by k. See Figure 10 for a description of the components of S(i,k). We can now formulate a definition of directionality in terms of the model parameters.

Definition 3.7: A Markov Random Field is isotropic at order i if b(i,1) = b(i,2). Otherwise, it is said to be anisotropic at order i. The notation b(i,.) implies isotropy at order i and signifies the common value of b(i,1) = b(i,2).

The notion of anisotropy agrees with our intuitive notion of directionality in textures. As we shall see in later chapters, the directionality is determined to some extent by the relationship between the b(i,1) and b(i,2).

3.5 Specific Models

We now discuss specific choices for the conditional distributions that are appropriate for use in image processing applications. A desirable model includes a small number of parameters and a close correspondence to the local image formation process. The two models which are considered in detail are the binary model and its extension, the binomial model. We mention a few others which are appropriate for image processing applications but have received little or no attention so far.

3.5.1 Binary Model

The variable X takes on the value 0 or 1. Define the variable T by the formula:

$$T = a + \sum_{k=1}^{2} \sum_{i=1}^{4} b(i,k)S(i,k) \qquad (3.11)$$

where the S(i,k) are composed of the sum of the values shown in Figure 10.

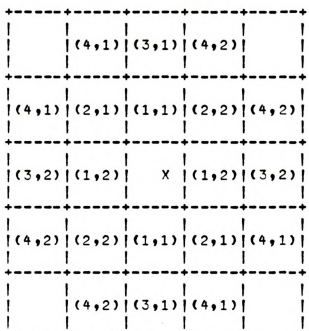


Figure 10 Constituents of the S(i,k). The points are labeled with the index of the S(i,k) in which they appear. For example, S(4,2) is the sum of 4 points, 2 near the lower left-hand corner and two near the uppper right-hand corner.

We obtain the conditional probability for X by:

$$(3.12)\quad p(X=x|T) = \frac{\exp(xT)}{1 + \exp(T)}.$$

3.5.2 Binomial Model

The binomial model generalizes the binary model to the case of variables with range {0,1,2,...,G-1}. We assume that the conditional probability p(X=k|T) is binomial with parameter θ(T), and number of tries G-1. The value of the parameter θ(T) is given by

$$(3.13)\quad \theta(T) = \frac{\exp(T)}{1 + \exp(T)}.$$

3.5.3 Other Auto-Models

We can use any member of the exponential family as the conditional distribution. Slight adjustments have to be made to the theory for the continuous case. Besag [10] cites the Poisson, exponential, and normal distributions as possible candidates.

It should be noted that the auto-normal scheme is not the same as the auto-regressive scheme mentioned in section 2.2.6. In the auto-normal formulation, the variables X(i) have a Gaussian distribution with common variance and mean T, where T is given in equation (3.11). The value of T depends on the neighborhood configuration about X(i).

3.5.4 Other Lattice Models

It is possible to relax assumptions (1) and (2) of section 3.2. The resulting models are still Markov Random Fields, but have a somewhat different conditional probability structure. The resulting fields have either too many parameters or are relatively intractable.

3.5.4.1 Strauss Model

Strauss [97] introduced a type of Markov Random Field that uses unordered 'colors' {0,1,2,...,C}. After some reduction, he obtains a conditional distribution of the form:

$$(3.14) \quad p(X=k|.) = \frac{\exp(T)}{1 + \exp(T)}, \quad k = 1,2,\ldots,C$$

$$p(X=0|\bullet) = 1 - \sum_{k=1}^{C} p(X=k|\bullet)$$

where

$$T = u + \sum_{i=1}^{C} v(i)N(X,i)$$

and u and v(i) are parameters and can be any real numbers. The notation p(X=k|•) represents the probability that X takes the value k, conditioned on its neighbors. The quantities N(X,i) represent the number of neighbors of the point X at level i.

This model is oriented toward cases where the levels are genuinely unordered, without arithmetic relations. It might be appropriate for detection of clustering among pixels that have been classified according to some land-use scheme for LANDSAT pictures. The v(i) parameters control the affinity for the ith color to cluster.

One difficulty with the above model is that the "left-over" color 0 (in the sense that there is no parameter v(0)) tends to have either a very small or very large conditional probability. Limited experiments showed that no natural, equalized picture could be fit to this model with four levels. It was thus not considered further as a texture model.

The most serious drawback of this model, even in the case where the unordered color assumption might be correct, is the number of parameters. Unless some simplifying assumptions can be made, one needs to add one parameter for each direction and for each gray level added. For example, a 32-gray level picture with first-order anisotropy would require 63 parameters under the Strauss model, yet only three under the binomial model. Of course, the figure of 63 can be reduced if there is equality of attraction, but the assumption of unordered colors tends to require a large number of parameters. Strauss was only interested in the case when all of the v(i) were the same, resulting in a much simpler estimation problem.

3.5.4.2 Welberry-Galbraith Fields

Work on another kind of binary Markov Random Field than presented earlier has been progressing since 1975 [38], [39], [101], [102], [103], [104], [104]. A Welberry-Galbraith field is a Markov Random Field which uses unilateral dependence to define the conditional probabilities. It is controlled by four parameters, a, b, c, and d, as shown in figure 11.

Boundary values are either random or determined by a distribution matching the true distribution. The model, although simple to explain and easy to simulate, has

Neighborhood Configuration

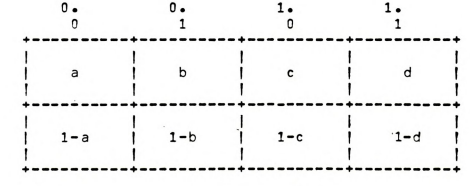


Figure 11 Conditional Probabilites for the Welberry-Galbraith Field: The probability of a 0 or 1 given the neighbors to the left and below. The four possible unilateral neighborhood configurations form the columns.

proven to be difficult to analyze. The expectation of the number of 1-valued points is known approximately along with the fact that the process is ergodic. This model is essentially the same as the Markov Mesh [1]. Also of interest is Pickard's [82] binary field.

3.6 Summary

We have defined a Markov Random Field formally and stated the main characterization result, the Hammersley-Clifford theorem. The general Markov Random Field model was reduced through simplifying assumptions to a compact form with a few parameters. The parameters define the order of the process along with the directionality. We specified in detail the models which will be investigated for texture generation and sythesis purposes, the binary and binomial models. Finally, we have mentioned some other models that may have some use in image processing applications.

CHAPTER 4. SIMULATION OF MARKOV RANDOM FIELDS

4.1 Introduction

In chapter 3 we gave the conditional probability formulation for Markov Random Fields. In order to generate textures that are the visual representation of Markov Random Fields, we need a procedure that yields a sample from a Markov Random Field with given parameters. Fortunately, such procedures exist and have been used extensively in physics to investigate the properties of two- and three-dimensional Ising lattices [16], [34], [32]. Simulation is usually needed to estimate many of the properties of Markov Random Fields since the analytical calculations are, for the most part, unsatisfactory [33].

4.2 Definitions

The required theory for the simulation of Markov Random Fields comes from the theory of discrete, finite-state Markov chains. We review the main results and definitions, mostly to fix notation. We follow Feller [31] and Hammersley [43] closely.

We consider discrete integer times, t = 1,2,3,... . The process has an at most countable state set S(1), S(2),... . Let X(t) denote the state of the process at time t.

Definition 4.1: The system of states {S(j)} is said to be a Markov chain if the conditional probability of the system being in a state at time t, given its state at all other times, depends only on the state it was in at time t-1. We may write this as:

p(X(t)=S(j)| All values of X(t)) =

p(X(t)=S(j)|X(t-1)=S(i)).

In all cases that we consider, the conditional probability depends only on the state value and is independent of time. In this case, the conditional probabilities are called stationary and we can write without ambiguity: p(X(t)=S(j)|X(t-1)=S(i)) = p(i,j). The p(i,j) are called the (stationary) transition probabilities. The system is called symmetric if p(i,j) = p(j,i).

Definition 4.2: The n-step transition probabilities are p(n;i,j) = p(X(t)=S(j)|X(t-n)=S(i)). The First-passage probabilities are

f(n;i,j) =

p(X(t)=S(j),X(t-1)≠S(j),...,X(t-n+1)≠S(j)|X(t-n)=S(i)).

The mean recurrence times are

$$m(i,i) = \sum_{n=1}^{\infty} n f(n;i,i).$$

The m(i,i) represent the expected time for the process to return to state i given that it was in state i at time t.

Definition 4.3: The states S(i) and S(j) are called mutually accessible if there exist integers m and n so that p(m;i,j) and p(n;j,i) are both non-zero. Mutually accessible states are said to belong to the same class. A system with only one class of states is said to be irreducible.

We can now delimit some important classes of states.

Definition 4.4: A state S(i) is called positive if m(i,i) is finite, null if m(i,i) is infinite. If p(n;i,i) is non-zero only when n is a multiple of d, the state S(i) is called periodic (of period d). If d = 1, it is called aperiodic. A state S(i) is called recurrent if

$$\sum_{n=1}^{\infty} f(n;i,i) = 1$$

otherwise, it is called non-recurrent.

It can be shown (Theorem 1, p391, [31]) that the states of the same class are all non-recurrent, or all positive, or all null, and all have the same period.

The set of numbers $\{q(j)\}$ is called a limiting distribution provided

$$\lim_{n\to\infty} p(n;i,j) = q(j) = 1/m(j,j)$$

4.3 Simulation Procedure

The simulation procedure is explained by three theorems: (4.1), (4.2), and (4.3). We want a Markov chain whose states are the set of colorings $\{\underline{X}\}$ with limiting distribution $\{p(\underline{X})\}$. We can sample such a chain and observe colorings $\{\underline{X}\}$ with frequency given by $\{p(\underline{X})\}$.

Theorem 4.1 gives sufficient conditions for a Markov chain to have a unique limiting distribution. Theorem 4.2 shows how to convert a relatively arbitrary Markov chain to one with limiting distribution $\{q(j)\}$. The key feature of theorem 4.2 is the fact that we need only know the ratios $\{q(j)/q(i)\}$ in order to obtain the desired chain. Finally, theorem 4.3 shows how to calculate the set of ratios $\{q(j)/q(i)\} = \{p(\underline{X})/p(\underline{Y})\}$ from the conditional distribution of a Markov Random Field without explicit calculation of the set of $\{p(\underline{X})\}$.

Theorem 4.1: Let {S(j)} be the states of an aperiodic, irreducible Markov chain. Let {q(j)} be a set of numbers that satisfy the following three conditions:

a. $q(j) > 0$ for all j

b. $$\sum_{j} q(j) = 1$$

c. $$\sum_{i} q(i)p(i,j) = q(j)$$

Under these conditions we have

1. Each state S(i) is positive

2. The set {q(j)} is unique is satisfying conditions a.,b.,c.

3. $$\lim_{n\to\infty} p(n;i,j) = q(j) = 1/m(j,j)$$

proof: See Feller (p393, [31]). □

Theorem 4.1 justifies the use of limiting distribution as a description of the set {q(j)}. We describe a procedure for transforming a given Markov chain with transition matrix P* = {p*(i,j)} to another chain with transition matrix P = {p(i,j)} such that the new chain has limiting distribution {q(j)}.

Theorem 4.2: Consider a symmetric, aperiodic, irreducible Markov chain with transition matrix P*. Let {q(j)} be a set of positive numbers with sum 1. Then the Markov chain with transition matrix P has limit distribution {q(j)}, where P is defined by:

$$p(i,j) = \begin{cases} p^*(i,j)q(j)/q(i) & \text{if } q(i) > q(j) \\ p^*(i,j) & \text{if } q(j) \geq q(i) \end{cases}$$

$$p(i,i) = p^*(i,i) + \sum_{j}{}' p^*(i,j)(1 - q(j)/q(i))$$

where the ' on the summation means summation over all indices j with q(j)/q(i) less than 1.

proof: See Hammersley [43]. □

To apply theorem 4.2 to the problem of generating a sample from a Markov Random Field, we need to identify the constituents of the Markov chain. The relevant pieces are:

State set: All colorings $\underline{X}$ of the lattice L.

Matrix P*: $p^*(i,j) = 1/z$, z is the total number of colorings.

$\{q(j)\}$: The limiting distribution value for the state $\underline{X}$ should be $p(\underline{X})$, where $p(\cdot)$ is the joint probability mass function over the set of all colorings.

In chapter 6, we will study the joint probability formulation of a Markov Random Field in detail. The expression is unwieldy but we do not actually need to use it, because the actual numbers $\{q(j)\}$ are never needed in the calculation of the $p(i,j)$ in theorem 4.2; we need only examine the set of ratios $\{q(j)/q(i)\} = \{p(\underline{Y})/p(\underline{Y})\}$. Following Besag [10], we have

Theorem 4.3: Let $\underline{X}$ and $\underline{Y}$ be two colorings of the Markov Random Field lattice L. Then:

$$\frac{p(\underline{Y})}{p(\underline{X})} = \prod_{i=1}^{M} \frac{p(X(i)=y(i)|X(1),X(2),\ldots,X(i-1),Y(i+1),\ldots Y(N))}{p(X(i)=x(i)|X(1),X(2),\ldots,X(i-1),Y(i+1),\ldots Y(N))}$$

proof: See Besag [10], p195. □

In general, we are interested in textures with the same number of pixels at each gray level. Such textures are the output of a histogram-flattening procedure such as the Equal Probability Quantizing algorithm described by Haralick [45]. It is therefore appropriate to limit our state set to those colorings $\underline{X}$ which have a uniform histogram. In practice, this is done by starting with an image that is generated by coloring the point (i,j) with level k, where k is chosen with equal probability from the set {0,1,2,...,G-1}. The convergence to the limit distribution is unaffected by the choice of initial configuration; only the rate at which equilibrium is reached depends on the choice of the initial configuration.

Given a state (i.e. coloring) $\underline{X}$, we choose the next state $\underline{Y}$ to be the same as $\underline{X}$ except that the gray values of two randomly selected points are interchanged. In the

notation of theorem 4.2, all $p^*(i,j)$ are equal, so we need only look at $q(j)/q(i) = p(\underline{Y})/p(\underline{X}) = r$. The algorithm is diagrammed in Figure 12. This algorithm was used by Flinn [33] and was invented by Metropolis et al. [73].

Any initial assignment of gray levels to the points of the image can be made. One possibility is to choose the next state $\underline{Y}$ as $\underline{X}$ with one pixel changed at random. This allows us to begin with any histogram whatsoever and reach states $\{S(i)\}$ with the desired frequency. The other side of this problem is that the choice of parameters of the Markov Random Field determines the expected histogram. This means that if we arrange to make the histogram to be uniform when, in fact, the parameters do not determine a Markov Random Field with a uniform histogram, the procedure will converge to a Markov Random Field with different parameters than intended. In a practical sense, this is not a significant problem since we usually want to generate a texture that resembles a given texture. We can generate a texture starting with exactly the same histogram as the given texture and use the measured parameters from the sample.

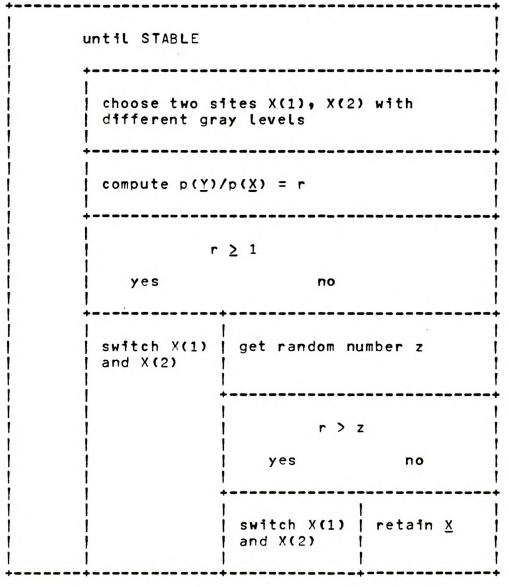


Figure 12 Algorithm for generating Markov Random Field with joint probability function p(X). The coloring Y is obtained from the coloring X by switching the values of the points X(1) and X(2).

4.4 Convergence Properties

Theorem 4.2 guarantees that the application of the algorithm in Figure 12 will eventually result in a lattice in a state $\underline{Y}$ with the frequency controlled by $p(\underline{X})$. The practical question is how long this will take.

We first need to define a time-dimension for the simulation. Suppose the lattice is N by N and let M=N**2. We consider M attempted exchanges or switches to constitute one *iteration*. Notice that this ignores attempted exchanges between pixels of the same color. We have experimental guidelines for the number of iterations required to achieve a lattice that matches the input parameters. In general, it was observed that in 10 iterations or less either the number of changes per iteration drops to one percent of M or the measured parameters match the input parameters within about 5 percent. These guideleines define the variable "STABLE" in Figure 12. On a PDP-11/34 computer, the time required for one iteration on a 64 by 64 image was two to three minutes depending on the number of gray levels.

We give an example of the convergence for a specific case. We want to simulate a first-order Markov Random Field with binary variables with parameters a = -2, b(1,.) = 1 on a 128 by 128 lattice. The notation follows equation (3.11) of chapter 3. The estimated parameters b(1,.) are shown plotted against number of iterations in Figure 13. The value of the parameter a is not shown because, as will be proven later, in the uniform histogram case, a is -2b(1,.). The graph flattens rather quickly and stays within 0.05 of the intended value of 1.0 for b(1,.). Figure 14 shows the number of changes observed per 256 attempted exchanges, as the estimates were made every 1/64 iteration. Thus, although in an iteration of M attempted exchanges we are still observing nearly M/2 changes, the chain is near equilibrium as the observed model parameters are within a small tolerance of the intended parameters.

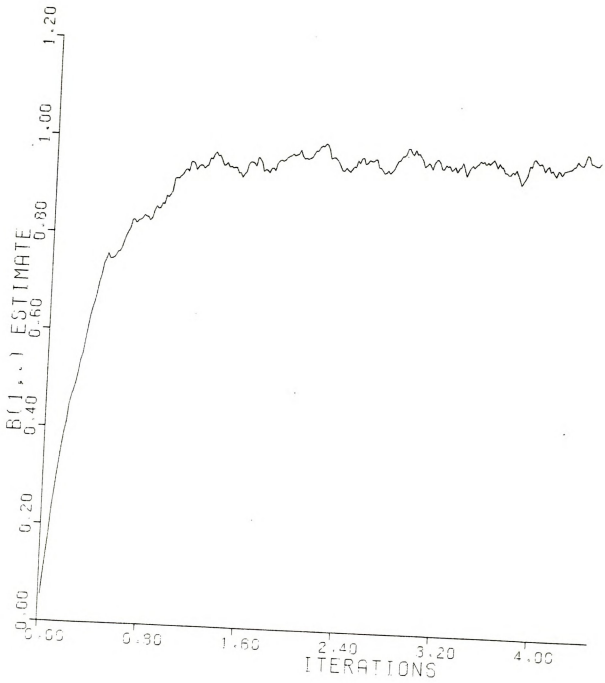


Figure 12: Convergence of b(1,.).

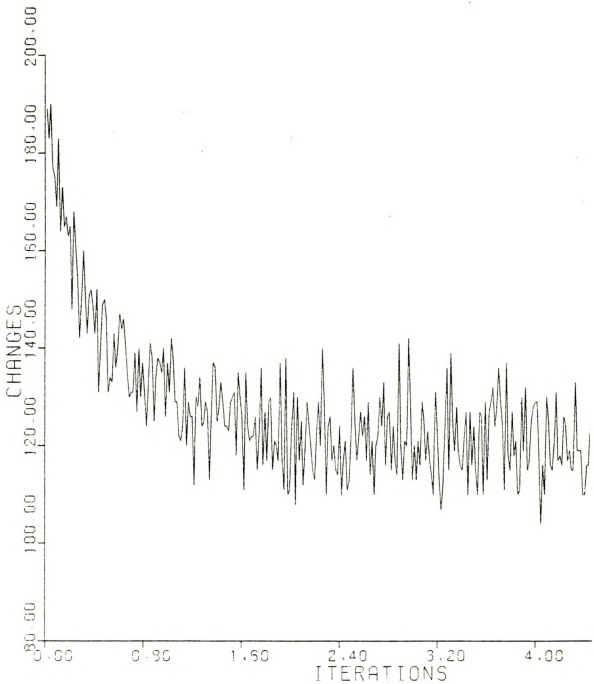


Figure 14: Behavior of Number of Exchanges.

4.5 Examples of Textures

We present some examples of textures generated according to various settings of Markov Random Field parameters. These images are representative of the kind of results that can be achieved but are not necessarily attempts to imitate real textures. They should rather be considered to be an 'alphabet' of Markov Random Field textures. In chapter 5, we exhibit generated textures matching observed textures.

(i) Clustering effects

Figure 15 shows a series of 64 by 64 binary textures with various degrees of clustering. This is an isotropic, first-order model as represented by equations (3.6) and (3.7). Figure 15a represents binary 'noise' in the sense that each pixel has probability .5 of being black and .5 of being white independently of all other pixels. In Markov Random Field terms, this means a is 0 and $b(1,1) = b(1,2) = 0$. The value of $b(1,1)$ is increased from 0 in Figure 15a to 3.5 in Figure 15h. The increase in clustering is clearly visible.

(ii) Anisotropic Effects

Figure 16 shows extreme anisotropy in first-order and second-order models on a 64 by 64 lattice. In Figure 16b, a = -2.0, b(1,1) = 1.93, and b(1,2) = .16. The quantity b(1,1) controls the horizontal clustering so there is a large amount of line-likeness in that direction. As b(1,2) is non-negative, a small amount of vertical clustering is present, resulting in thickened and noisy horizontal lines. Contrast this with Figure 16a, which has parameters b(1,1) = -2, and b(1,2) = 2.1. The value of b(1,2) causes vertical clustering, similar in intensity to the horizontal clustering of Figure 16b. The significant difference is that the negative value of the parameter b(1,1) forces 'clean' vertical lines with virtually no horizontal clustering.

The decidedly diagonal effect of Figure 16c results from the use of a second-order structure. The clustering in the NW-SE direction is pronounced since the parameter in this direction is 1.9 while the parameters in all other directions are quite small.

(iii) Ordered Patterns

Many of the applications of the Ising model involve studying the checkerboard-like patterns obtained with negative clustering parameters. In this sense, a perfect checkerboard is ordered and represents a limit state for a an alloy or magnet [77]. This is illustrated by Figure 17, which has b(1,1) = -2.25, b(1,2) = -2.16 on a 64 by 64 lattice. The most likely configuration is a black pixel surrounded by four white pixels or vice versa.

(iv) Attraction-Repulsion Effects

An attraction-repulsion process involves having low-order parameters positive resulting in clustering but high-order parameters negative in order to inhibit the growth of clusters. If high-order parameters were also positive, large clusters would result, whereas negative high-order parameters yields small clusters.

Figure 18 shows the effect of anisotropic clustering with inhibition. The first-order parameters of Figure 18a are approximately 0, which accounts for an immediate visual impression of randomness. On closer examination, there are many short horizontal and vertical lines, but

very few diagonal joins. The lack of diagonal joins is a result of negative second-order diagonal parameters. In Figure 18b, the first-order clustering parameters b(1,1) and b(1,2) have been increased, resulting in longer horizontal and vertical lines.

Figure 19 shows two isotropic attraction-repulsion textures. These are the result of positive first and second-order parameters and negative third and fourth-order parameters. As a consequence of the high-order inhibition, the cluster size is relatively smaller than one would expect if the third- and fourth-order parameters were zero.

(v) Multiple Gray Scale Textures

The binary textures above illustrate the essential features of the visual attributes of a Markov Random Field but are fundamentally unrealistic as textures. We now turn our attention to the binomial model. We follow the notation of section 3.5.

It should be noted that many of these images appear blurry and out of focus. This effect is not due to the reproduction process but is intrinsic to the model. If

there is no inhibition (via negative high-order parameters), then the binomial model tends to have smooth transitions from black to white. The binomial distribution is unimodal and, as a consequence, values above and below the mean gray value are highly probable also. This results in a tapering of the gray scale around maxima and minima. Such a tapering as one moves away from black or white points has an effect similar to a neighborhood averaging or low-pass filter.

Figure 20 shows a 4-gray level picture with considerable clustering. Figure 20a is isotropic and first-order, whereas Figure 20b is second-order anisotropic with diagonal clustering. Figures 21a and 21b represent typical multiple gray scale pictures with isotropic fourth-order clustering.

Figure 22 shows a pattern similar to Figure 16, but with 32 gray levels. The resemblance to wood-grain is apparent. Figures 23a and 23b show the result of attraction-repulsion processes with multiple gray levels. Figure 23a has the appearance of reticulated photographic film due to strong third and fourth-order inhibition. The diagonality in Figure 23b is a result of strong repulsion in some directions and clustering in others.

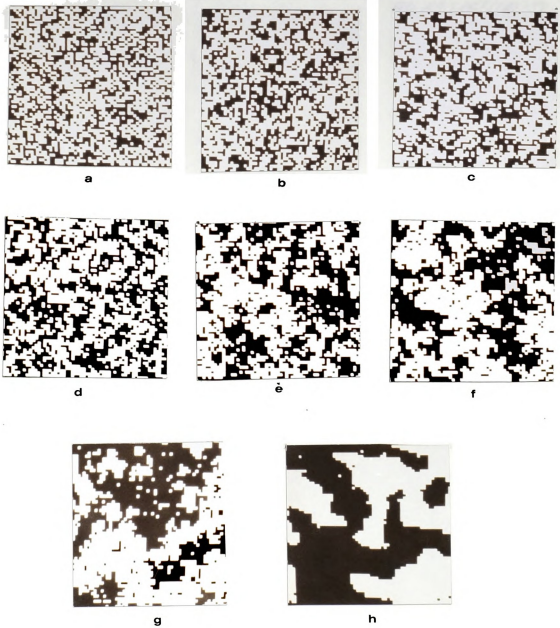


Figure 15: Isotropic First Order Textures. The b(1,.) parameters are: (a) 0.0, (b) 0.50, (c) 0.75, (d) 1.1, (e) 1.26, (f) 1.52, (g) 1.79, (h) 3.0. In all cases, the a parameter is -2b(1,.).

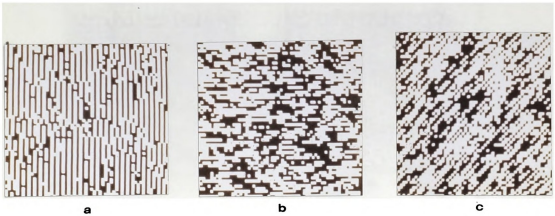


Figure 16: Anisotropic Line Textures. The parameters are (a) a= -0.26, b(1,1) = -2., b(1,2) = 2.1, b(2,1) = 0.13, b(2,2) = 0.015. (b) a= -2.04, b(1,1) = 1.93, b(1,2) = 0.16, b(2,1) = 0.07, b(2,2) = 0.02. (c) a = -1.9, b(1,1) = -0.1, b(1,2)= 0.1 , b(2,1)= 1.9 , b(2,2)= -0.075.

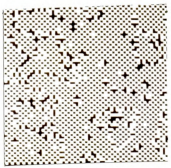
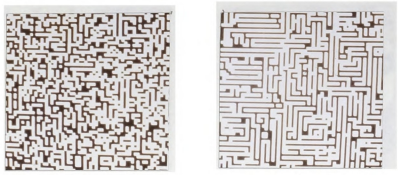
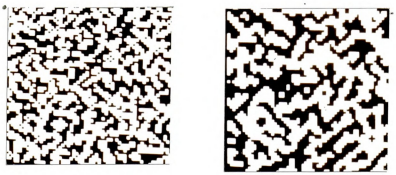


Figure 17: Ordered Pattern. The parameters are a = 5.09, b(1,1) = -2.25, b(1,2) = -2.16.



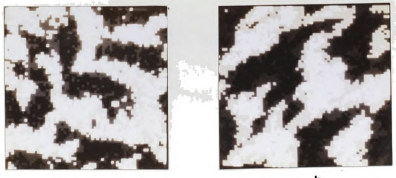
a b

Figure 18: Diagonal Inhibition Textures. The parameters are (a) a = 2.19, b(1,1) = -0.088, b(1,2) = -0.009, b(2,1) = -1, b(2,2) = -1. (b) a =0.16, b(1,1) = 2.06, b(1,2) = 2.05, b(1,2) = -2.03, b(2,2) = -2.10.



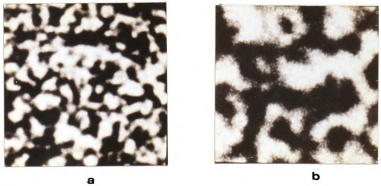
a b

Figure 19: Isotropic Inhibition Textures. The parameter values are: (a) a =-0.97, b(1,.) = 0.94, b(2,.) = 0.94, b(3,.) = -0.42, b(4,.) = -0.49. (b) a = -4.6, b(1,.) = 2.62, b(2,.) = 2.17, b(3,.) = -0.78, b(4,.) = -0.85.



a b

Figure 20: Clustered Textures with 4-Gray Levels. The parameters are (a) a = -2, b(1,.) = 1.0. (b) a = -2, b(1,.) = 1.0, b(2,1) = 1.0 b(2,2) = 0.0.



a b

Figure 21: Clustered Textures with 16 and 32 Gray Levels. The parameters are (a) 16 levels, a = -2.0, b(1,.) = b(2,.) = b(3,.) = b(4,.) = 0.05. (b) 32 levels, a = -2.0, b(1,.) = b(2,.) = b(3,.) = b(4,.) = 0.05.

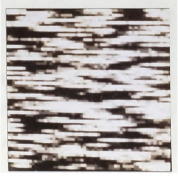


Figure 22: Horizontal Texture with 16 Gray Levels. The parameters are a = -2.0, b(1,1) = 0.08, b(1,2) = 1.0.

a b

Figure 23: Attraction-repulsion Textures with 16 Gray Levels. The parameters are (a) a = -2, b(1,.) = b(2,.) = .2, b(3,.) = b(4,.) = -0.1. (b) a = -2, b(1,.) =0.2, b(2,1) = -2.2, b(2,2) = .2, b(3,1) = -0.05, b(3,2) = 0.05, b(4,1) = -0.05, b(4,2) = 0.05.

4.6 Summary

We have reviewed the notion of a Markov chain and seen how to obtain a new Markov chain with desired limit distribution from a relatively arbitrary chain. This construction led directly to an algorithm for the generation of Markov Random Fields with specified parameters. An example of the convergence of the procedure was given, along with some guidelines on the number of iterations required to obtain a Markov Random Field with the desired parameters. Finally, we presented a catalog of textures generated according to various settings of the model parameters.

CHAPTER 5. MODELLING OF NATURAL TEXTURES

5.1 Introduction

In previous chapters, we have discussed the probabilistic structure of Markov Random Fields and have shown how samples from Markov Random Fields can be generated. One of the principal contributions of this thesis is the implementation of a statistical measure of the correspondence between an observed texture and a texture model. No prior study has performed this kind of evaluation. All prior studies in texture modelling have considered a model adequate if its parameters yielded good classification in pattern recognition experiments or if it was found to be the best-fitting among a number of models tested. For example, Deguchi and Morishita [26] determine the best size for a neighborhood in an auto-regressive scheme, but do not give any overall guidelines on when the auto-regressive scheme fits the observed texture.

We first explain the method used to estimate the Markov Random Field parameters and perform hypothesis tests. This is followed by the results of testing textures from the Brodatz texture album [17] for a fit to various Markov Random Field texture models. The final

section shows textures generated synthetically in an effort to match natural textures.

5.2 Estimation of Parameters

Following the notation of equations (3.12) and (3.13), this section explains the estimation of the parameter set {b(j,k)} from a textured sample using the binomial model. The binary model is a special case of the binomial model and is not handled separately.

The technique used to estimate the parameters is Maximum Likelihood Estimation. Let p(X|.) denote the conditional probability p(X=x| Neighbors of X), where X is a point of the lattice L. The usual log likelihood is given by

$$(5.1) \qquad l = \sum_{X} \ln(p(X|.))$$

where the summation extends over all points of the lattice.

It is difficult to maximize L as stated in equation (5.1) since the summands are not independent. Besag [9] provided a solution to this problem. Instead of forming L as a sum over all points of the lattice, the lattice is partitioned into disjoint sets of points called codings. Each coding is chosen so that its points are independent. This can be done by adequately spacing the X points so that if X(i) and X(j) are two points in a coding, then X(i) is not a neighbor of X(j) in the Markov Random Field sense.

The number of codings required depends on the order of the process. We would like each coding to be as large as possible because the larger the coding, the more samples are available to estimate the parameters. A first-order process requires at least two codings for estimation purposes, as shown in figure 24. A second-order process requires spacing so that three by three neighborhoods do not interfere. This yields four codings as shown in Figure 25. Third- and fourth-order processes require separation of three units since they are based on five by five neighborhoods, as shown in Figure 26. There are nine codings in this case.

1	2	1	2	1	2	1	2	1	2
2	1	2	1	2	1	2	1	2	1
1	2	1	2	1	2	1	2	1	2
2	1	2	1	2	1	2	1	2	1
1	2	1	2	1	2	1	2	1	2
2	1	2	1	2	1	2	1	2	1
1	2	1	2	1	2	1	2	1	2
2	1	2	1	2	1	2	1	2	1
1	2	1	2	1	2	1	2	1	2
2	1	2	1	2	1	2	1	2	1

Figure _24_ First-Order Coding Pattern. There are two codings labelled 1 and 2.

1	2	1	2	1	2	1	2	1	2
3	4	3	4	3	4	3	4	3	4
1	2	1	2	1	2	1	2	1	2
3	4	3	4	3	4	3	4	3	4
1	2	1	2	1	2	1	2	1	2
3	4	3	4	3	4	3	4	3	4
1	2	1	2	1	2	1	2	1	2
3	4	3	4	3	4	3	4	3	4
1	2	1	2	1	2	1	2	1	2
3	4	3	4	3	4	3	4	3	4

Figure 25 Second-Order Coding Pattern. There are four codings labelled 1, 2, 3, 4.

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

Figure 26 Third- or Fourth-Order Coding Pattern. There are nine codings labelled 1-9.

The actual estimation procedure is straightforward. Let l(i) be the log likelihood for the ith coding:

$$(5.2)\quad l(i) = \sum_{X'} \ln(p(X'|\cdot))$$

where the summation extends over all points X' in the coding i. In equation (5.2), p(X'|·) depends on the order of the Markov Random Field whose parameters are being estimated.

We seek to maximize l(i). This is done by finding values of the parameters a and {b(j,k)} so that

$$(5.3)\quad \partial l(i)/\partial a = 0 \text{ and } \partial l(i)/\partial b(j,k) = 0$$

for j = 1,2,...,r and k = 1,2 where r is the order of the process. If the process is isotropic at some order j, then we only consider derivatives with respect to b(j,·) in equation (5.3).

The system of equations (5.3) is solved numerically by the usual multivariable extension of Newton's method [52]. We omit the routine expression for the derivatives

in equation (5.3) and the second partials that are needed in Newton's method. The stopping criterion for the iteration process was fifteen iterations, or a change in the magnitude of successive parameter estimate vectors of less than 10E-06. Except for a few cases among the natural and synthetic texture samples, ten iterations sufficed to provide a residual of less than 10E-06. This is fortunate, since each iteration of Newton's method in the multivariate case effectively requires the inversion of a matrix of size equal to the number of parameters, which could be as many as nine for fourth-order anisotropic textures.

An estimate of the parameter vector is obtained for each coding. Our final estimate of the parameters is the average value over all the codings. On a PDP-11/34 computer an unoptimized estimation procedure requires thirty seconds per coding for binary textures and eighty seconds per coding for eight gray level textures. These timings are averages for textures of size 64 by 64.

5.3 Hypothesis Testing

Note that we really have only one sample from the unknown distribution $p(\underline{X})$ on the set of colorings of the

lattice L. From the conditional probability point of view, each observed configuration of neighbors and the value of the center point X is a sample. In this sense, we have M = N**2/k samples of the conditional density p(X|.) for the N by N lattice L, where k is the number of codings. We can then perform a chi-square test of the fit between the expected frequencies for each center pixel and the observed frequency. The expected frequencies are computed using the estimated parameters with an appropriate reduction in degrees of freedom.

An example of a chi-square test is shown in Figure 27. The null hypothesis is:

H0: The texture is a sample from a Markov Random Field with the estimated parameter set {a,b(j,k)}

while the alternative hypothesis is simply the negation of H0. The estimated parameters for the example shown in Figure 27 are a = -4.26, b(1,1) = 2.70, and b(1,2) = 1.5. The number of cells is G, the number of gray levels, times the number of observed neighborhood configurations. The example is a first-order, anisotropic binary lattice, so there are 2*9 cells possible and all were observed. The entries in the table are of the form 'observed(expected)' and the expected entry is computed using the conditional

probability distribution with the estimated parameters. The degrees of freedom are computed by means of the formula:

(5.4) df = (G-1)*Nc - E

where G is the number of gray levels, Nc is the number of neighborhood configurations (in Figure 27, each neighborhood configuration is a row), and E is the number of estimated parameters. In the example of Figure 27, there are two gray levels, nine neighborhood configurations and three estimated parameters, so we have (2-1)*9-3 = 6 df. We also use the convention of having at least one expected observation per cell. This results in a reduction in the number of cells and degrees of freedom.

Since we are performing a number of tests on the same data (one on each of k codings), there is a great likelihood of having the hypothesis of a fit to a Markov Random Field scheme accepted on some codings and rejected on others. The results of the tests are not independent. If they were independent, and we performed k tests at a level α, then the probability of no rejections would be $(1-\alpha)^{**}k$.

S(1,1)	S(1,2)	X = 0	X = 1
0	0	625(621.•)|	5(9.•)|
0	1	240(244.•)|	20(16.•)|
0	2	14(17.•)|	8(5.•)|
1	0	79(80.•)|	18(17.•)|
1	1	93(91.•)|	85(87.•)|
1	2	21(14.•)|	52(59.•)|
2	0	4(3.•)|	8(9.•)|
2	1	12(17.•)|	244(239.•)|
2	2	8(8.•)|	512(512.•)|

Figure 27 Example of a Chi-Square Test. This is a binary 64 by 64 texture. A first-order anisotropic scheme was fitted. The results of coding number 1 are shown above. Chi-square is 12.30957, on 6 df.

Besag [10] suggests a conservative solution to the problem of ambiguous results. Suppose the most significant result we observed over k codings was exactly at level p. The overall significance of the set of tests is then taken to be kp. For example, in a first-order isotropic scheme we obtained two chi-squared values of 7.78 (p=0.10) and 11.3 (p=0.01) on 3 df. The value 11.3

has exact significance level of 0.01. Since there are two codings, we take our overall significance level to be exactly 0.02 = 2*0.01, and would reject the null hypothesis at the 5 percent level.

In the tables in section 5.5, we have given the number of "conservative" rejections at the five percent level for each scheme along with the number of samples of each texture that had 0,1,...,k rejections over k codings at the five percent level. This second tabulation leads to a "liberal" rejection policy of rejecting a Markov Random Field fit if any coding is rejected.

5.4 Description of the Data

The study of natural textures was based on twelve pictures from the Brodatz texture album [17]. The plates used are given in Table 1. Each plate was photographed on 35mm film to form 24mm by 36mm slides. The slides were digitized in such a way that the small dimension occupied slightly over 256 pixels of the 480 by 640 image of Spatial Data Eyecom system. The 256 by 256 images were split into sixteen non-overlapping 64 by 64 subimages and also into four 128 by 128 non-overlapping subimages. The gray scale of each subimage was reduced from 256 gray

levels to both two and eight gray levels using Equal Probability Quantizing [45].

Table 1: Textures Used in the Study. The plate numbers refer to the Brodatz texture album [17].

Name	Plate Number
Brick Wall	D94
Ceiling Tile	D86
Pressed Cork	D4
Calf Fur	D93
Grass Lawn	D9
Handmade Paper	D57
Pebbles	D31
Beach Sand	D29
Straw Screening	D49
Water	D38
Wood Grain (1)	D69
Wood Grain (2)	D70

5.5 Evaluation of the Fit

The textures were evaluated for their fit to various models. The purpose of this analysis was twofold. First, we need to validate that the Markov Random Field model is generally applicable to textured images. The second objective is to formulate some general guidelines on how to choose an appropriate model, in terms of order and the degree of anisotropy and isotropy, to generate specific types of textures.

5.5.1 Binary Texture Results

Except for the screen texture, some first- or second-order model was able to give at least ten acceptances, under the previously mentioned conservative decision rule, for each texture sample. Detailed analysis of the screen texture samples showed very few distinct neighborhood configurations. This is a consequence of its regularity. The few neighborhood configurations dominate the computation of the Markov Random Field parameters. However, the low frequency neighbor probabilities are not properly controlled by these parameters, resulting in large chi-square values. Essentially, the histogram is so skewed toward these configurations that the positivity

condition of section 3.2 is nearly violated.

As an example of this, Figure 28 consists of 64 repetitions of a random sixteen by sixteen pattern. It was analyzed on the basis of first- and second-order isotropic and anisotropic models. Because of the repetitions, there are very few different configurations of neighbors. Only 42 different ones appear out of a possible 81 in a second-order anisotropic scheme. Chi-squared values are in the range 1000-3000 on 20-40 degrees of freedom, depending on the model. The estimated parameters vary wildly from coding to coding. For example, on a second-order anisotropic scheme, the parameter a was estimated as 0.02, 0.8, -0.243, 0.325. The b(1,1) estimates were 0.376, -0.469, +0.469, and -0.567. Similar variability was present in all the parameters.

Tables 2 through 9 give the results of testing binary texture samples under a number of models. A test against a first-order model results in two hypothesis tests, one for each coding. There are sixteen subimages of each texture and each may be rejected on zero, one, or two codings at the five percent level. The number of subimages which had each number of rejections is recorded

in the last three columns of tables 2 and 3. Also listed is the number of conservative acceptances, of which there are a maximum of sixteen possible. For an appreciation of the difference between the conservative and the liberal rule, compare the number of conservative acceptances with the number of subimages which had no rejections. Tables 4 through 9 parallel the format of tables 2 and 3 except that they represent tests performed using the second-order codings. There is a possibiity of zero, one, two, three, or four rejections per subimage, as there are four codings.

Besag [10] makes the point that one cannot compare the fit of a second-order model to the fit of a first-order model unless the same coding scheme is used in both cases. From Tables 2 and 3, it would appear that the first order scheme fits very well, but in many cases it is inferior to a second-order scheme when considered on the same coding. Table 10 gives the best results on second-order coding for each of the binary textures. The general good fit of the first order model should be reconciled with the fact that there are only two codings rather than the four for a second-order scheme, which means that fewer rejections are likely.

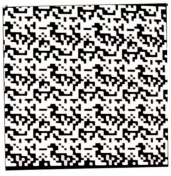


Figure 28: Non-Markovian Periodic Texture. This is a 64 by 64 section of a 128 by 128 texture. The basic pattern is 16 by 16 and is repeated 64 times over the image.

We have limited our attention to the case of 64 by 64 textures. Although third-order estimates can be made which give good visual results in texture generation experiments explained later, we cannot reliably perform a chi-square test of them on the 64 by 64 lattice. The number of cells with only one member is very large since the number of possible cells with a fully anisotropic third-order model is 729 while there are only about 585 in a single third-order coding (there are 9 codings in all).

Our preferred choices for the best-fitting model are based on a simple rule: choose the model that gives the largest number of acceptances with a conservative decision rule. The results of this rule are displayed in table 10. We would not consider a fit to be adequate unless the majority of samples from the texture class fit the model.

Table 2: Isotropic, First-Order, Binary Tests. The parameters estimated are a, b(1,.).

Texture Name	Conservative Acceptances	Rejections at 0.05 level 0	1	2	3	4
Brick	14	10	6	0		
Ceiling Tile	10	9	4	3		
Cork	14	12	3	1		
Fur	5	2	11	3		
Grass	15	12	3	1		
Paper	16	12	4	0		
Pebbles	14	10	6	0		
Sand	15	14	1	1		
Screen	0	0	2	14		
Water	10	9	7	0		
Wood (1)	9	8	8	0		
Wood (2)	9	6	9	1		

Table 3: Anisotropic, First-Order, Binary Tests. The parameters estimated are a, b(1,1), b(1,2).

Texture Name	Conservative Acceptances	Rejections at 0.05 level 0	1	2	3	4
Brick	11	11	5	0		
Ceiling Tile	11	6	7	3		
Cork	11	10	5	1		
Fur	6	5	8	3		
Grass	13	11	4	1		
Paper	13	13	3	0		
Pebbles	11	9	5	2		
Sand	15	12	3	1		
Screen	1	0	5	11		
Water	8	8	7	1		
Wood (1)	9	7	4	5		
Wood (2)	4	2	9	5		

Table 4: Isotropic, First-Order, Binary Tests, on second-order spacing. The parameters are a, b(1,.).

Texture Name	Conservative Acceptances	Rejections at 0.05 level 0	1	2	3	4
Brick	16	13	3	0	0	0
Ceiling Tile	16	11	4	1	0	0
Cork	14	12	3	1	0	0
Fur	11	8	3	5	0	0
Grass	15	11	4	1	0	0
Paper	15	14	2	0	0	0
Pebbles	14	10	6	0	0	0
Sand	14	12	4	0	0	0
Screen	0	0	1	1	5	9
Water	11	10	2	3	1	0
Wood (1)	13	10	6	0	0	0
Wood (2)	13	6	9	1	0	0

Table 5: Anisotropic, First-Order, Binary Tests, on second-order spacing. The parameters are a, b(1,1), b(1,2).

Texture Name	Conservative Acceptances	Rejections at 0.05 level 0	1	2	3	4
Brick	9	6	8	2	0	0
Ceiling Tile	15	11	4	1	0	0
Cork	12	10	4	2	0	0
Fur	10	9	3	3	1	0
Grass	16	11	4	1	0	0
Paper	14	10	6	0	0	0
Pebbles	11	9	1	6	0	0
Sand	14	10	5	1	0	0
Screen	2	0	2	3	6	5
Water	11	9	6	1	0	0
Wood (1)	12	7	7	2	0	0
Wood (2)	11	3	10	3	0	0

Table 6: First-Order (Isotropic), Second-Order (Isotropic) Binary Tests. The parameters are a, b(1,.), b(2,.).

Texture Name	Conservative Acceptances	Rejections at 0.05 level 0	1	2	3	4
Brick	5	1	3	7	3	2
Ceiling Tile	12	9	4	2	1	0
Cork	11	6	6	4	0	0
Fur	12	7	8	1	0	0
Grass	13	12	3	1	0	0
Paper	11	8	6	2	0	0
Pebbles	11	6	6	2	2	0
Sand	14	9	5	2	0	0
Screen	0	0	3	0	2	11
Water	12	7	5	3	1	0
Wood (1)	10	5	8	3	0	0
Wood (2)	10	7	6	1	2	0

Table 7: First-Order (Isotropic), Second-Order (Anisotropic) Binary Tests. The parameters are a, b(1,•), b(2,1), b(2,2).

Texture Name	Conservative Acceptances	Rejections at 0.05 level 0	1	2	3	4
Brick	5	3	6	4	3	0
Ceiling Tile	13	10	5	1	0	0
Cork	14	11	5	0	0	0
Fur	10	7	8	1	0	0
Grass	15	11	5	0	0	0
Paper	13	12	4	0	0	0
Pebbles	11	9	4	0	2	1
Sand	14	12	4	0	0	0
Screen	3	3	1	1	11	0
Water	13	11	4	1	0	0
Wood (1)	13	9	5	2	0	0
Wood (2)	12	11	3	1	0	0

Table 8: First-Order (Anisotropic), Second-Order (Isotropic) Binary Tests. The parameters are a, b(1,1), b(1,2), b(2,.).

Texture Name	Conservative Acceptances	Rejections at 0.05 Level 0	1	2	3	4
Brick	2	0	7	4	3	2
Ceiling Tile	12	11	2	1	2	0
Cork	9	3	6	6	1	0
Fur	9	7	7	2	0	0
Grass	14	9	4	2	1	0
Paper	11	6	4	2	4	0
Pebbles	10	9	2	3	1	1
Sand	12	11	4	1	0	0
Screen	4	2	2	5	1	6
Water	13	7	9	0	0	0
Wood (1)	10	7	6	3	0	0
Wood (2)	13	5	9	1	1	0

Table 9: First-Order (Anisotropic), Second-Order (Anisotropic) Binary Tests. The parameters are a, b(1,1), b(1,2), b(2,1), b(2,2).

Texture Name	Conservative Acceptances	Rejections at 0.05 level 0	1	2	3	4
Brick	6	4	7	3	2	0
Ceiling Tile	13	10	4	2	0	0
Cork	14	11	4	1	0	0
Fur	11	8	6	2	0	0
Grass	15	10	4	2	0	0
Paper	14	11	4	1	0	0
Pebbles	12	11	2	1	1	1
Sand	14	14	2	0	0	0
Screen	5	3	5	3	2	3
Water	14	13	3	0	0	0
Wood (1)	14	10	3	3	0	0
Wood (2)	15	13	2	1	0	0

Table 10: Best-fitting Binary Texture Models. In the table below, 'I' means that the fewest rejections were obtained by using isotropic estimation, while 'A' signifies the best results were obtained using anisotropic parameters. The symbol '---' in the second-order column signifies that the best results were obtained using a first-order model.

Texture Name	First Order	Second Order	Acceptances
Brick	I	---	16
Ceiling Tile	I	---	16
Cork	I	A	14
Fur	I	I	12
Grass	A	---	16
Paper	I	---	15
Pebbles	I	---	14
Sand	A	A	14
Screen	A	A	5
Water	A	A	14
Wood (1)	A	A	14
Wood (2)	A	A	14

5.5.2 Binomial Texture Results

With the experience gained from the binary fits, we limited our attention to four samples from each of the eight gray level pictures. First-order models yielded consistently negative results. The binomial model is unable to effectively model bimodal or uniform conditional probabilities. The binomial model always has a peak at exactly one value for any choice of θ. Also, if there are two likely gray values which are not contiguous, then no choice of the θ parameter can yield the correct probabilities.

As in the binary case, third-order analysis cannot be performed on samples of size 64 by 64 for eight gray level textures. In the matching experiments, estimation was performed for third-order textures using 128 by 128 samples. Even this is barely enough, and results in about thirty percent collapsing of the cells when the cells are pooled to force expectations of at least one per cell. A first-order model with anisotropy can have as many as 225 cells, while a full third-order model with anisotropy can use as many as 225**3 cells. Of course, the number of cells is limited by the number of points on the lattice.

Tables 11 and 12 give the results of fitting eight-color models to four samples from each texture class. The format is analagous to Tables 2 through 9, except that only four samples per texture class were used. As in the case of the binary textures, good results were obtained for all but the inhomogeneous textures: Water, Wood, and Pebbles. The binomial model has difficulty in handling large areas of equal brightness. All of the textures which did not fit the model well are either blotchy or regular, like the image of the screen. Fine-grained textures can be handled and, as we shall see, generated by the binomial model easily. The best results of the two sets are shown in Table 13.

Table 11: Eight gray level, First-order(Anisotropic), Second-order(Anisotropic). The parameters are a, b(1,1), b(1,2), b(2,1), and b(2,2).

Texture Name	Conservative Acceptances	Rejections at 0.05 level				
		0	1	2	3	4
Brick	0	0	0	0	0	4
Ceiling Tile	2	1	3	0	0	0
Cork	4	3	1	0	0	0
Fur	1	1	2	0	1	0
Grass	3	3	1	0	0	0
Paper	4	3	1	0	0	0
Pebbles	0	0	0	0	0	4
Sand	4	4	0	0	0	0
Screen	0	0	0	0	0	4
Water	0	0	0	1	0	1
Wood (1)	1	0	0	0	0	4
Wood (2)	0	0	0	0	0	4

Table 12: Eight gray level, First-order(Isotropic), Second-order(Anisotropic). The parameters are a, b(1,.), b(2,1), and b(2,2).

Texture Name	Conservative Acceptances	Rejections at 0.05 level 0	1	2	3	4
Brick	3	0	2	2	0	0
Ceiling Tile	3	3	1	0	0	0
Cork	1	1	0	1	1	1
Fur	4	4	0	0	0	0
Grass	3	1	2	1	0	0
Paper	3	2	2	0	0	0
Pebbles	0	0	0	0	0	4
Sand	4	2	2	0	0	0
Screen	0	0	0	0	0	4
Water	0	0	0	0	0	4
Wood (1)	0	0	0	0	0	4
Wood (2)	0	0	0	0	0	4

Table 13: Best-fitting Eight-gray level Texture Models. In the table below, 'I' means that the fewest rejections were obtained by using isotropic estimation, while 'A' signifies the best results were obtained using anisotropic parameterss. The symbol '---' indicates that neither model was appropriate.

Texture Name	First Order	Second Order	Acceptances
Brick	A	I	3
Ceiling Tile	A	I	3
Cork	A	A	4
Fur	I	A	4
Grass	A	A	4
Paper	A	A	4
Pebbles	---	---	0
Sand	A	A	4
Screen	---	---	0
Water	---	---	0
Wood (1)	A	A	1
Wood (2)	---	---	0

5.6 Texture Matching Experiments

This section examines the viability of the Markov Random Field as a supervised texture generation procedure. The input texture is measured using the Maximum Likelihood approach described in section 5.3. The results of that evaluation are used as the input to the generation procedure in Figure 12.

As mentioned in section 5.5, the third-order model on a 64 by 64 texture cannot be reliably fitted since it causes too many empty or near-empty cells. Limited experimentation showed that if the image size was increased to 128 by 128, a sensible chi-square estimate could be made, though the parameters did not change very much from the estimates made on the 64 by 64 textures.

5.6.1 Binary Textures

Binary textures have far simpler structures than multi-gray level textures. If they are not regular, like a tiling of the plane by polygons, then we can describe their general appearance by a few general characteristics. This is really an abbreviated list of the intuitive textural attributes defined in the first chapter.

(i) Directionality: Vertical, Horizontal, Diagonal, or None.

(ii) Cluster size: Large, Medium, or Small

(iii) Homogeneity: Homogeneous or Inhomogeneous. Inhomogeneous images have different characteristics in different parts of the picture.

(iv) Special Features: Regularity, Line-likeness, Blob-likeness.

Table 14 illustrates rough characterization on the basis of one 64 by 64 sample of the features given above. We can use this guide to evaluate our success in matching the generated textures. Of course, we would like to be able to say that one texture "looks like" another, but we need some way of quantifying this correspondence. Figure 29 shows the results of generating the twelve textures from the estimated parameters. The parameters used to generate the synthetic textures were obtained by averaging the parameter estimates from each of the codings of a single subimage. The choice of subimage was arbitrary. A third-order estimate was used in all cases except for the pictures of Wood grain(1) and Pebbles. In these two

Table 14: Binary Texture Characteristics

Texture	Direction	Cluster Size	Homo-geneity	Special Features
Bricks	Vertical	Large	Low	Regular
Ceiling	Vertical	Small	High	None
Cork	Diagonal	Small	High	None
Fur	None	Large	Low	None
Grass	Diagonal	Small	High	None
Paper	Diagonal	Small	High	None
Pebbles	None	Large	Low	Blobs
Sand	None	Medium	High	None
Screen	Vertical	Small	High	Lines, dots
Water	Vertical	Medium	Low	None
Wood(1)	Horiz.	Large	Low	Blobs
Wood(2)	Horiz.	Large	Low	Blobs

cases, a first-order model gave better visual results. First-order models tend to form in blob-like aggregations and cannot correctly characterize fine-structured textures. Moreover, they are inadequate in showing any directionality except vertical and horizontal.

The clear failures are:

Bricks: The regularity and neat rectangles are not present. Cluster size is close to correct along with the overall vertical structure. The regularity occurs at an order of around 20 pixels, and there is little chance of capturing such a structure from a 64 by 64 picture.

Fur,Wood (1), Wood (2): The inhomogeneity is missed in the synthetic examples. What remains of these pictures after binary quantization can hardly be called a texture. The images are estimated to be fine-structured rather than clean, blob-like shapes like pebbles, because their noise component modifies the estimate so that third-order inhibition is used in the generation. This is an unfortunate consequence of the estimation procedure. It is possible that a preliminary smoothing might help in getting a more correct estimate. In all cases, the directionality is correctly simulated in these textures.

The remaining nine textures are reasonably approximated by the simulated textures. In the pictures of Cork, Grass, and Paper, diagonality is the overriding feature and this is correctly modelled. The screen image is remarkably similar to the original. The third-order repulsion effect provides the curious checkerboard effects along the lines in both the original and the generated texture. Without this inhibition effect, the image would resemble the vertical images shown in Figure 16.

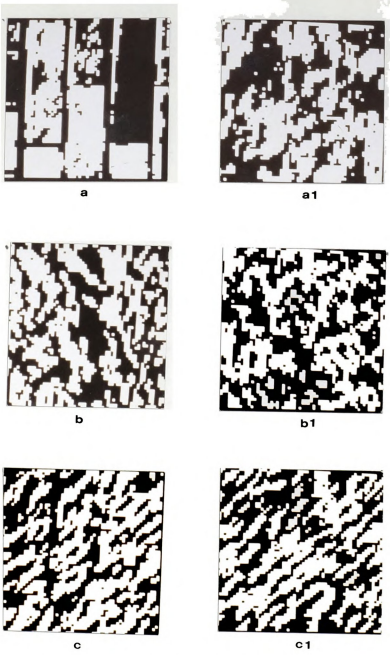
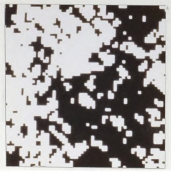
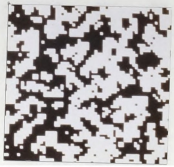


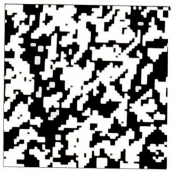
Figure 29 Real and Synthetic Binary Textures: (a) Binary Bricks (a1) Synthetic Binary Bricks (b) Binary Ceiling tile (b1) Synthetic Binary Ceiling Tile (c) Binary Cork (c1) Synthetic Binary Cork



d

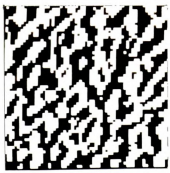


d1



e

e1



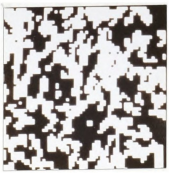
f

f1

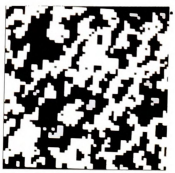
Figure 29 Continued: (d) Binary Fur (d1) Synthetic Binary Fur (e) Binary Grass (e1) Synthetic Binary Grass (f) Binary Paper (f1) Synthetic Binary Paper

g

g1

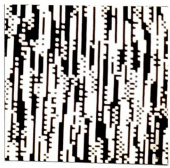


h



h1

i

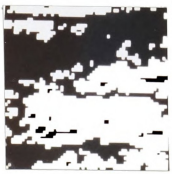


i1

Figure 29 Continued: (g) Binary Pebbles (g1) Synthetic Binary Pebbles (h) Binary Sand (h1) Synthetic Binary Sand (i) Binary Screen (i1) Synthetic Binary Screen

j

j1



k

k1

L

L1

Figure 29 Continued: (l)Binary Water (l1)Synthetic Binary Water (k) Binary Wood Grain(1) (k1) synthetic Binary Wood Grain(1). (L) Binary Wood Grain(2) (L1) Synthetic Binary Wood Grain(2).

5.6.2 Binomial Texture Matching

The same basic data set of Brodatz [17] textures was used in the binomial matching experiments as was used in the binary experiments. The pictures were quantized to have eight gray levels using histogram equalization. However, during the estimation of the third-order Markov Random Field parameter set, it was found that far too many cells were empty or contained only a single member when the image size was 64 by 64. Therefore, a 128 by 128 image size was used for both estimation and matching. In all cases, a third-order model was estimated and used to generate the synthetic textures.

Numerically, the Markov Random Field parameters of the estimated binomial textures are very similar, which accounts for the similar appearance of the generated textures. We have omitted some of the obvious failures: Water, Wood Grain (1), Wood Grain (2), and Pebbles. When considered as an eight-gray level image, these textures take on a distinctly inhomogenous appearance. At this size, they look like pictures of objects, whereas the generated textures look like a fine-grain field. The Markov Random Field always results in a homogeneous covering of the image, which cannot result in a blotchy

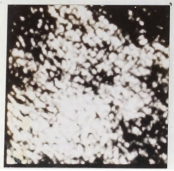
image unless the parameters are extreme. As an example, the fur pictures, Figure 30a and 30a1, show the result of an inhomogeneity in the image.

The other clear failure is the Bricks picture. As in the binary case, the Bricks image has a regular structure. The Markov Random Field can only detect a hint of a vertical structure. This is shown in Figures 30b and 30b1.

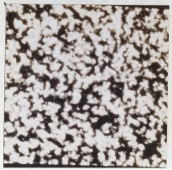
Ceiling tile, Cork, Grass, Paper, and Sand are handled adequately. Missing are the large black holes in the synthetic ceiling tile picture, but there are some dense black patches. The distinctly three-dimensional appearance of the the handmade paper, admittedly a tactile property, is not captured either.

The Screen texture has the general line-like quality, but lacks the straightness present in the original. This could be corrected, but the estimated parameters of the screen texture do not support the further inhibition needed to straighten out the lines and keep them narrow rather than fuzzy.

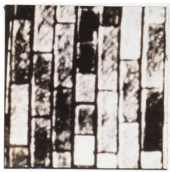
The model seems to be adequate for duplicating the micro-textures, but is incapable of handling strong regularity or cloud-like inhomogeneities. These experiments should be taken as an exploration of the limits of a purely statistical approach to texture without any a priori knowledge at all. For example, if we knew that the Bricks picture was supposed to have rectangles of a certain size and orientation, then we could start with them as an outline and then fill in the rectangles with a Markov Random Field.



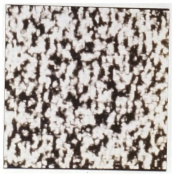
a



a1

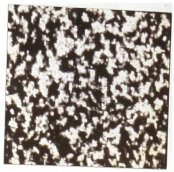


b



b1

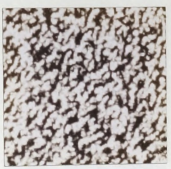
c



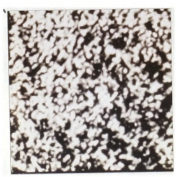
c1

Figure 30: Real and Synthetic Eight Level Textures (a) Fur (a1) Synthetic Fur (b) Bricks (b1) Synthetic Bricks (c) Ceiling Tile (c1) Synthetic Ceiling Tile

d



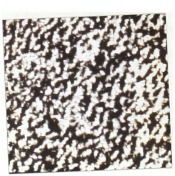
d1



e

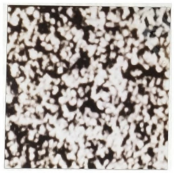
e1

f



f1

Figure 30 Continued: (d) Cork (d1) Synthetic Cork (e) Grass (e1) Synthetic Grass (f) Paper (f1) Synthetic Paper



g

g1

h

h1

Figure 30 Continued: (g) Sand (g1) Synthetic Sand (h) Screen (h1) Synthetic Screen

5.7 Summary

We have explained the estimation procedure for the Markov Random Field parameters. Hypothesis testing was presented as a method of objective evaluation of the fit between a texture model and a texture. We computed the fit between Markov Random Field textures and a collection of samples from the Brodatz texture album [17]. Experiments in the generation of textures that matched real images were performed, with some successes and explainable failures.

CHAPTER 6. CO-OCCURRENCE MATRICES

6.1 Introduction

Features based on co-occurrence matrices continue to play an important role in texture analysis [24], [88]. In a model-based approach, we would like to determine the distribution of these features as a function of the model parameters. If we knew these distributions, then we could, at least in principle, determine the discriminative capability of classification algortithms based on the co-occurrence features. As an example, let $\underline{W}(1)$ and $\underline{W}(2)$ be two distinct parameter vectors for a texture model. Suppose further that the mean, $S = E[\sum X(i)]$, is the same for textures from either the class defined by $\underline{W}(1)$ or the one defined by $\underline{W}(2)$. Then any classification scheme based solely on S would be unable to discriminate between textures from the two classes.

A study along these lines was performed by Conners and Harlow [24]. They created textures using a one-dimensional Markov process. The Markov process specifies the probability that a pixel with gray level i follows a pixel with gray level j in a row. The rows of the image are uncorrelated. This texture model provides a

method of computing co-occurrence matrices from the transition matrix of the process. Texture algorithms can be compared on their capability of distinguishing textures with different transition matrices.

The difference between the above study and prior comparative studies [106] is that Conners's study did not compare the texture analysis algorithms on the basis of their performance on a texture database, but rather predicted their performance on texture classes defined by model parameters. Empirical studies certainly are of interest to users of texture analysis algorithms, but they guide in the selection of textural recognition algorithms only insofar as the results of the empirical study can be generalized to new data.

Our purpose in this chapter is to begin a theoretical calculation of textural feature distributions based on Markov Random Field parameters. We limit our attention to the case of binary, first-order Markov Random Fields.

6.2 Joint Probability Formulation

This section defines the variables we need to calculate the co-occurrence matrices. We assume that the binary variables take values in {-1,1}. Following Bartlett [7], we have the following expression for the joint probability mass function:

$$p(\underline{X}) = Z\exp(-\alpha S(\underline{X}) - Y(1)U1(\underline{X}) - Y(2)U2(\underline{X})) \tag{6.1}$$

where

$$S(\underline{X}) = \sum_{(r,s)} X(r,s) \tag{6.2}$$

$$U1(\underline{X}) = \sum_{(r,s)} X(r,s)X(r-1,s) \tag{6.3}$$

$$U2(\underline{X}) = \sum_{(r,s)} X(r,s)X(r,s-1). \tag{6.4}$$

The lattice is assumed to have a periodic boundary.

The quantity Z in equation (6.1) is called the partition function. It depends on α, $\gamma(1)$, and $\gamma(2)$ and is a normalizing constant so that $p(\underline{X})$ is a valid probability mass function.

The quantity $E[S(\underline{X})]$ is called the mean of the process. If the value of α is zero, then $E[S(\underline{X})]$ is also zero. This is not hard to show. For each coloring $\underline{X}$, let $-\underline{X}$ denote the negative of $\underline{X}$, in the sense that the value of each point $X(r,s)$ of the lattice L is reversed in sign. Then $S(\underline{X}) = -S(-\underline{X})$, but $U1(\underline{X}) = U1(-\underline{X})$, and $U2(\underline{X}) = U2(-\underline{X})$. Since the quantity α is zero, we see from equation (6.1) that $p(\underline{X}) = p(-\underline{X})$. Hence, we conclude that

$$E[S(\underline{X})] = \sum_{\underline{X}} S(\underline{X})p(\underline{X})$$

$$= \sum_{\substack{\underline{X} \text{ with} \\ S(\underline{X})>0}} (S(\underline{X}) + S(-\underline{X}))p(\underline{X})$$

$= 0$

The case of $\alpha = 0$ is of interest in image processing applications. As mentioned earlier, we usually reduce the gray scale from some high level like 256 to 16 or less in such a way that the numbers of pixels present at each gray level are equal. If we reduce the gray scale to two levels, the observed $S(\underline{X})$ should be close to 0 in the {-1,1} formulation.

If we transform the variables so they take values in {0,1} instead of {-1,1}, then the conditional probability parameters of section 3.5.1 can be written as:

(6.5) $a = 2\alpha - 4\gamma(1) - 4\gamma(2)$

$b(1,i) = 4\gamma(i),\ i = 1, 2.$

When $\alpha = 0$, $a = -b(1,1) - b(1,2)$ (in the isotropic case, we have $a = -2b(1,\cdot)$). This means that an anisotropic, first order texture appears to have three parameters: a, b(1,1), and b(1,2). If we force the mean to be N**2/2, i.e. $E[S(\underline{X})] = 0$ in the {-1,1} formulation, then there are really only two parameters: b(1,1) and b(1,2). Strauss [97] properly considers the a parameter to be a "nuisance" parameter, while the b(j,k) parameters control

the clustering in the lattice.

6.3 Join Count Calculations

This section derives the formulas for the join count statistics, which in turn imply the co-occurrence matrices. We assume that $\alpha = 0$, i.e. the equalized histogram case.

The quantities U1 and U2, as defined in equations (6.3) and (6.4), are the join counts. By definition, the bivariate cumulant generating function for U1 and U2 is

$$K(\phi(1),\phi(2)) = \log(E[\exp(\phi(1)U1 + \phi(2)U2]).$$

Let $Z(\ (1),\ (2))$ denote the partition function for the case $\alpha = 0$. The peculiar form of the joint probability mass function in (6.1) implies that

$$(6.6)\quad K(\phi(1),\phi(2)) = \log Z(Y(1),Y(2)) - \log Z(Y(1)-\phi(1),Y(2)-\phi(2)).$$

From equation (6.6), we obtain

$$(6.7)\quad E[U1] = (\partial K(\phi(1),\phi(2))/\partial\phi(1))\big|_{\phi(1)=\phi(2)=0}$$

$$= \partial \log Z(\gamma(1),\gamma(2))/\partial\gamma(1)$$

and

$$(6.8)\quad E[U2] = \partial \log Z(\gamma(1),\gamma(2))/\partial\gamma(2).$$

It is thus possible to compute the expectation and moments (or even approximate the distribution by a cumulant expansion) of U1 and U2. In this approach, we require knowledge of the partition function Z. Onsager [79] calculated the exact value of Z for the case $\alpha = 0$. This is is only case for which analytical results are available. The somewhat unwieldy exact Z can be closely approximated by the more compact form given by Kac and Ward [61]. When N is large and even, and the lattice has a periodic boundary then we can write

$$(6.9)\quad \log Z(\gamma(1),\gamma(2)) = N(\cosh(\gamma(1))+\cosh(\gamma(2)) + \sum_{s=1}^{N}\sum_{r=1}^{N/2} \log H(r,s)$$

where

$$H(r,s) = (1+x^{**}2)(1+y^{**}2)$$

$$+ \; 2y(x^{**}2-1)\cos(2\pi r/N)$$

$$+ \; 2x(y^{**}2-1)\cos(2\pi s/N)$$

and $x = -\tanh(Y(1))$, $y = -\tanh(Y(2))$.

The expression for the expectation of U1 or U2 is

$$E[U_f] = N\tanh(Y(f)) + \sum_{s=1}^{N/2} \sum_{r=1}^{N/2} \frac{1}{H(r,s)} \frac{\partial H(r,s)}{\partial Y(f)} \tag{6.10}$$

and the variances are given by

$$\mathrm{Var}[U_f] = N\mathrm{sech}^2 Y(f) + \sum_{s=1}^{N} \sum_{r=1}^{N/2} \frac{1}{H(r,s)} \frac{\partial^2 H(r,s)}{\partial Y(f)^2} - \frac{1}{H^2(r,s)} \frac{\partial H(r,s)}{\partial Y(f)} \tag{6.11}$$

We are now in a position to compute the co-occurrence matrices. We need to define the join counts for the case of variables in the form {0,1}. Each lattice point X(r,s) has a left-hand neighbor X(r-1,s) and a lower neighbor X(r,s-1), assuming the boundary is periodic. We can then define the following set of numbers, in a manner consistent with current practice [80]:

BB1 = number of times a black pixel has a black left-hand neighbor.

WW1 = number of times a white pixel has a white left-hand neighbor.

BW1 = number of times a black pixel has a white left-hand neighbor.

WB1 = number of times a white pixel has a black left-hand neighbor.

The quantities BB2, WW2, BW2, and WB2 are defined in the same way except that "left-hand" is replaced with "lower." It is clear that on an N by N lattice we have

(6.12) BB1 + WW1 + BW1 + WB1 = N**2

and

(6.13) BB2 + WW2 + BW2 + WB2 = N**2

The co-occurrence matrices can be expressed, at least for displacements of (0,1) and (1,0), in frequency form from equations (6.12) and (6.13). On a torus, the (-1,0) and (1,0) co-occurrence matrices are the same. Their common form is:

(6.14) $$C(1,0) = \begin{bmatrix} BB1 & BW1 \\ WB1 & WW1 \end{bmatrix}$$

The (0,1) and (0,-1) co-occurrence matrix is:

(6.15) $$C(0,1) = \begin{bmatrix} BB2 & BW2 \\ WB2 & WW2 \end{bmatrix}$$

To compute these quantities from U1 and U2, observe that

(6.15) U1 = BB1 + WW1 - BW1 - WB1

(6.16) U2 = BB2 + WW2 - BW2 - WB2.

Combining equation (6.12) with (6.15) and equation (6.13) with (6.16), we obtain

(6.17) BB1 + WW1 = (N**2 + U1)/2

(6.18) BB2 + WW2 = (N**2 + U2)/2.

The final step in the computation rests on two assumptions. The first assumption is that the number of BB joins is closely approximated by the number of WW joins. Intuitively, this means that if the black pixels are clustered, the white pixels must also be clustered. The following proposition and its corollary establish that, from a Markov Random Field point of view, the clustering parameters for white and black are the same.

Proposition 6.1: Suppose a first-order Markov Random Field has conditional parameters a, b(1,1), and b(1,2). Then the Markov Random Field obtained by reversing each coloring of each realization has parameter set {a*,b*(1,1),b*(1,2)}, where

$$\begin{aligned} a^* &= -a - 2b(1,1) - 2b(1,2), \\ b^*(1,1) &= b(1,1), \\ b^*(1,2) &= b(1,2). \end{aligned}$$

Proof: Let p denote the conditional probability distribution of the original field and p* the conditional probability of the field obtained by reversing realizations. Let S*(1,1) and S*(1,2) denote the neighbor

sums in the reversed lattice (the quantities {S(i,k)} were defined in section 3.5.1). Then we have,

$$p(X=0|S(1,1),S(1,2))$$

$$= p^*(X=1|S^*(1,1)=2-S(1,1),S^*(1,2)=2-S(1,2)).$$

This means, following equations (3.11) and (3.12), that

$$\frac{1}{(1 + \exp(a + b(1,1)S(1,1) + b(1,2)S(1,2))}$$

$$= \frac{\exp(a^* + b^*(1,1)(2-S(1,1)) + b^*(1,2)(2-S(1,2))}{1 + \exp(a^* + b^*(1,1)(2-S(1,1)) + b^*(1,2)(2-S(1,2))}.$$

After some simplification, we obtain the relation that

$$(6.19)\quad 0 = a + b(1,1)S(1,1) + b(1,2)S(1,2)$$

$$+ a^* + b^*(1,1)(2-S(1,1)) + b^*(1,2)(2-S(1,2))$$

Equation (6.19) actually represents nine equations depending on the three values of S(1,1) and the three values of S(1,2). The only consistent solution is $a^* = a - 2b(1,1) - 2b(1,2)$, $b^*(1,1) = b(1,1)$, $b^*(1,2) = b(1,2)$. □

As an immediate consequence, in the equalized histogram case we can be more specific:

Corollary 6.1: If a binary, first-order Markov Random Field has parameter $\alpha=0$, then the parameters of the reversed field are the same as the parameters of the original field.

Proof: $a^* = -a - 2b(1,1) - 2b(1,2)$. But in the equalized histogram case $a = -b(1,1) - b(1,2)$ so $a^* = -a + 2a = a$. □

The second assumption needed to complete the computation is that the number of BW joins is closely approximated by the number of WB joins. Intuitively, this means that the conditional probability of a black pixel having a white neighbor is the same as the conditional probability of a white pixel having a black neighbor. This is a consequence of the equal clustering between black and white pixels, along with approximately equal numbers of white and black pixels.

Combining all of this, we obtain the final version of the co-occurrence matrices in frequency form:

$$(6.20)\quad C(1,0) = \begin{bmatrix} \frac{N^{**}2 + U1}{4} & \frac{N^{**}2 - U1}{4} \\ \frac{N^{**}2 - U1}{4} & \frac{N^{**}2 + U1}{4} \end{bmatrix}$$

$$(6.21)\quad C(0,1) = \begin{bmatrix} \frac{N^{**}2 + U2}{4} & \frac{N^{**}2 - U2}{4} \\ \frac{N^{**}2 - U2}{4} & \frac{N^{**}2 + U2}{4} \end{bmatrix}$$

6.4 Experimental Results

In order to test the adequacy of the approximations inherent in equations (6.20) and (6.21), we simulated a number of Markov Random Fields with various parameters and measured the correspondence between the expected and observed co-occurrence matrices.

The first set of simulations involves 200 samples of 64 by 64 isotropic Markov Random Field textures with b(1,.) parameters in the range 0 to 1. The high values closely approximate natural textures. A scatter-plot of the results of the expected and observed results appears in Figure 31. The horizontal axis is the clustering

parameter b(1,.). The vertical axis is the value of the c(0,0) entry (probability of a black-black join) in the C(1,0) co-occurrence matrix. The curve drawn in is a plot of the expected value of the c(0,0) entry plotted as function of b(1,.), computed using equation (6.20).

The second set of simulations involves a series of anisotropic textures with parameters b(1,1) and b(1,2) varying independently over the range 0 to 1.0 by increments of .1. Each texture was estimated under the first-order anisotropic model. The values of the C(0,1) and the C(1,0) co-occurrence matrices were recorded for each texture. The estimated parameters of the Markov Random Field were used to compute U1 and U2 using equation (6.10). The co-occurrence matrices were then approximated using equations (6.20) and (6.21).

The difference between the observed value of the probability of a black-black join and the approximated value is shown below in Table 15. There is a total of 128 co-occurrence matrices (two per texture). The mean error between the expected and observed black-black probability is 0.0135 over the 128 matrices. The average error is about three percent with a maximum error of eight percent. This error is entirely reasonable: the variance of the

join count in the isotropic case when b(1,.) is .75 is 84.6, with an expectation of 1211.

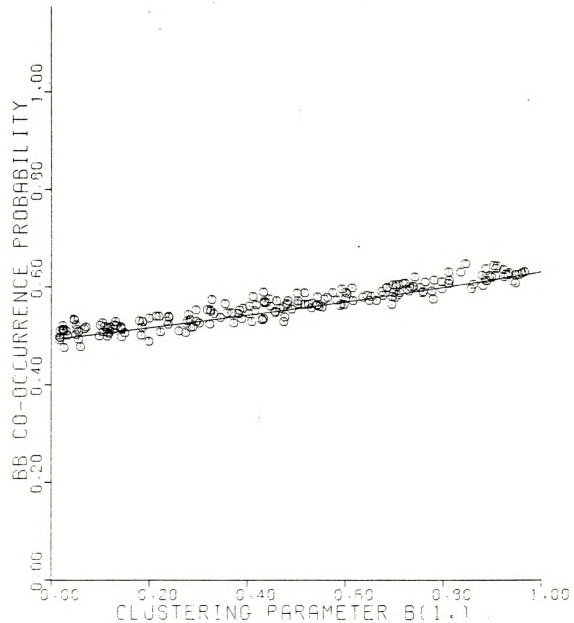


Figure 31: Expected and Observed Co-occurrence Probabilities.

Table 15: Errors in Co-occurrence Predictions. The error column represents the difference between the predicted black-black join probability and the observed.

Error	Frequency
Greater than 0.04	2
0.03 to 0.04	10
0.02 to 0.03	20
0.01 to 0.02	30
0.005 to 0.01	25
0.001 to 0.005	26
Less than 0.001	6
Total	128

6.5 Summary

We have formulated the joint probability form for binary Markov Random Fields. The partition function was shown to be the cumulant generating function for the horizontal and vertical join counts. Co-occurrence statistics were computed from the join counts. Finally, experiments showed good correspondence between the predicted co-occurrence matrices and the observed on a number of simulations.

CHAPTER 7. CONCLUSIONS AND DISCUSSION

7.1 Summary

Texture is defined as a stochastic, possibly periodic, two-dimensional image field. We reviewed the important aspects of texture and discussed vision research applicable to texture. The significant applications of texture analysis and synthesis were explained: classification, image segmentation, realism in computer graphics, and picture encoding.

A texture model was defined as a mathematical procedure capable of producing and describing a textured image. The models which have been used were summarized in some detail. Included were discretized continuous models, time-series, fractals, random mosaics, syntactic models, linear models, and Markov models. Each was found to be appropriate for some but not all types of textures.

The focus of this research was on the Markov Random Field model for texture, with special attention to the selection of an appropriate conditional distribution for the points of a lattice. The binomial model, where each point in the texture has a binomial distribution with

parameter θ controlled by its neighbors and "number of tries" equal to G, the number of gray levels, was taken to be the basic model for all the analysis. This represents the first use of the binomial model for image analysis and seems to be the first application of the binomial model for any purpose whatsoever.

A method of generating samples from a binomial or binary model was given. The colorings $\underline{X}$ of a lattice were identified with the states of a Markov chain. Convergence of the generation procedure was taken to mean that the equilibrium distribution of the Markov chain was the same as the set of probabilities $\{p(\underline{X})\}$. The practical aspect of the rate of convergence was discussed and an example was given to show that a Markov Random Field image with desired parameters is obtained rather quickly.

The Markov Random Field parameters control the strength and direction of the clustering of the image. Samples of various types of textures were exhibited to show the effects of various input parameters. These included line-likeness, blob-likeness, coarseness, and directionality. The power of the binomial model to simulate blurry or sharp images was demonstrated. Numerous types of images were found to require an

attraction-repulsion effect in the sense that the low-order Markov Random Field parameters were positive, causing clustering, while high-order parameters were negative, resulting in a fine structure of small clusters.

In order to determine the relevance of the binomial model to real textures, samples from the Brodatz texture album [17] were digitized. Overall, the micro-textures, grass, sand, cork, ceiling tile, and paper, obeyed the Markov Random Field model well. The use of a hypothesis test for the objective evaluation of the correspondence of the model to the sample was a major feature of this investigation.

Using the estimation procedure for Markov Random Fields, the parameters of the natural textures were measured. The measured parameters were used as input to the texture generation procedure. This was an attempt to see how far the statistical approach alone could be carried in the absence of any structural information about the texture. Micro-textures were successfully generated while the regular and inhomogeneous textures bore little resemblance to their synthetic counterparts.

An investigation was begun of the connection between the distribution of the usual statistics used for texture analysis and the Markov Random Field model. The co-occurrence matrices of binary, first-order Markov Random Fields were computed as a function of the field parameters. Good correspondence was noted between the co-occurrence matrices of sample textured images and the matrices predicted from their Markov Random Field parameters.

7.2 Discussion

This section is divided into two parts. The first part cites the advantages of the Markov Random Field image model and the second part details the limitations and defects of the model.

7.2.1 Advantages

The positive aspects of the Markov Random Field model are given below. Comparisions are made with other texture models.

(i) The model is fully two-dimensional and does not assume a causal (unilateral) dependence. Many texture models are one-dimensional, such as time series [66] or simple Markov scan-line textures [24]. It is essential, at least in principle, that a texture model be capable of describing spatial interaction over the plane in all directions. In fact, the Markov Random Field can even be adapted for use on irregular or triangular lattices by application of the Hammersley-Clifford theorem [10].

(ii) The texture parameters are measurable from samples and the appropriateness of the model can be assessed objectively by a hypothesis test. Contrast this with the texture generation process of Yokoyama and Haralick [109]. Their growth model represents a pattern generation process which certainly adds to our understanding of images, but we have no way of testing the sample image for correspondence to the model. Moreover, the Markov Random Field model allows the fitting of the sample texture directly to the model parameters. The current state of most models requires indirect fits by using variograms and correlation matches [90], [26].

(iii) The model parameters themselves are sufficient to generate images. The joint probability mass function $p(\underline{X})$ is sufficiently peaked so that textured image samples governed by it resemble each other visually. In a purely feature-based approach, we do not have the capability to generate images from features because the features do not uniquely define a texture class.

(iv) The pattern formation process, though specified locally, implies a global pattern. The consistency conditions enforced by the Markov Random Field cause a pattern over the entire lattice. As Besag explains in the discussion of his lattice model paper, [10]:

> Incidentally, the fact that a scheme is formally described as "locally interactive" does not imply that the patterns it produces are local in nature (cf. the extreme case of long-range order in the Ising model).

(v) The patterns formed are realistic. The binomial model allows natural smooth peaks and valleys in the gray level height field over the plane. Sharp or blurry images can be controlled by the amount of high-order inhibition. Contrast this to the Poisson checkerboard with uniformly colored blocks and sharp boundaries [75]. If the assumption of uniform cell coloring is dropped from the

mosaic models, then their parameters cannot be estimated by currently available means.

(vi) The patterns generated by varying the model parameters can be studied and classified. The results of chapter 4 show the visual significance of the parameter settings. Directionality, coarseness, gray level distribution, and sharpness can all be controlled by choice of the parameters.

(vii) The parameter estimation procedure can be implemented in a parallel algorithm. Each parameter estimate is performed on disjoint codings, each of which can be processed separately and then averaged to form the final result. Even the hypothesis tests could be done in parallel.

7.2.2 Disadvantages and Limitations

The Markov Random Field model has the following disadvantages, most of which are common to any purely stochastic approach to texture:

(i) Regular textures are not modelled very well. The only successful regular textures produced were the one-pixel wide checkerboard, Figure 17, or the screen, Figure 29i. The screen texture is really a one-dimensional texture consisting of highly correlated rows. The Markov Random Field approach is unable to provide the texture of regular cloth, a regular tiling, or the highly repetitive texture shown in Figure 29.

(ii) The textural primitives of the Markov Random Field are non-geometric. The shape of the cells in a Markov Random Field has a distribution determined by the parameters. We cannot enforce any geometrical conditions or force things as triangular shapes to appear, such as is possible in a bombing model [90]. We can only control the clustering characteristics: size, aspect ratio, density, and orientation.

(iii) Large images are required to get good parameter estimates. It would appear that third-order estimates require at least 128 by 128 pictures for eight gray level textures. When the gray scale is two or four, 64 by 64 seems to be enough. Low cell frequencies discourage the use of the chi-square test in these cases. In fact, the chi-square test is really based on the variance and some

other type of test may be more appropriate for testing third-order statistics.

(iv) The theoretical properties of the model are, in general, difficult to obtain. For example, the theoretical auto-correlation is known only numerically in the first-order binary case [33] and even less is known for higher-order models unless some stringent simplifying assumptions are made like b(1,.) = b(2,.) [28].

7.3 Suggestions For Future Research

Listed below are areas for investigation that extend the work in this thesis. All use the Markov Random Field model in some way.

(i) Obtain numerically (or analytically) the autocorrelation and higher-order co-occurrence matrices. This is neeeded to complete the theoretical study begun in chapter 6. Special attention is needed for the case of more than two gray levels.

(ii) Expand the data base of texture samples. The Brodatz texture samples certainly do not exhaust the possibilities of the model. It may be possible, for

example, to use the Markov Random Field model as a method of modelling film grain noise [36]. The binomial model textures have a distinctly biomedical flavor and may be of use in modelling images obtained by microscopy.

(iii) For the binary case, it is possible to apply the estimation procedure to the case of pictures thresholded at other than the fifty percent point of the histogram. Rosenfeld [88] has begun a productive investigation using the twenty-five percent point as a threshold. The clusters are smaller and carry more of the the structure of the picture without enormous, randomly linked clusters.

(iv) Consider other conditional distributions. Besag [10] lists auto-normal and auto-poisson as possible candidates.

(v) Try the full second-order model, as expressed in equation (3.9). This may provide sufficient information about the image without any need to go as far as the third-order. The full second-order model has only four codings, which is a distinct computational advantage.

(vi) Image encoding experiments can be performed in a manner similar to those performed by Modestino and Fries [75]. The appropriate block size for the reconstructed mosaic could be as small as 32 by 32. It is conceivable that even a smaller size could be used as long as the process does not require high-order parameters. One of the problems in generating entire images with a single set of parameters is that the image generated is homogeneous, but the target image is not. Using a mosaic approach could compensate for this.

(vii) Use Welberry-Galbraith type Markov Random Fields as explained in section 3.5.4.2.

(viii) Apply the Markov Random Field model in a hierarchical manner. First, estimate the parameters on the given resolution. Next, consider the lattice of dimension half the original dimension by either averaging non-overlapping two by two neighborhoods or simply selecting every other point. The reduced lattice parameters are then estimated and the process is repeated to get a sequence of parameter vectors which may describe the structure of the image better than a single set.

(ix) Texture segmentation experiments can be performed by dividing the lattice into 32 by 32 non-overlapping blocks and estimating the Markov Random Field parameters on each block. It is possible to perform a hypothesis test for equality of parameters [10] and this could be used to determine the region membership.

(x) A test of homogeneity is needed. The Markov Random Field model assumes homogeneity in the image. Some verification of this should be made before estimating the parameters and assessing the goodness-of-fit.

LIST OF REFERENCES

LIST OF REFERENCES

[1] Abend,K., Harley,T.J., and Kanal,L.N., 'Classification of Binary Random Patterns,' IEEE Transactions on Information Theory, IT-11, 1965, 538-544.

[2] Ahuja,N., 'Connectivity in Lattices and Mosaics,' Proceedings of the 4th International Joint Conference on Pattern Recognition, Kyoto, Japan, November, 1978, 488-493.

[3] Ahuja,N., Mosaic Models for Image Analysis and Synthesis, Ph.D. dissertation, Department of Computer Science, University of Maryland, College Park, Maryland, 1979.

[4] Akaike,H., 'Fitting Autoregressive Model for Prediction,' Annals of the Institute of Mathematical Statistics, 21, 1969, 243-247.

[5] Akaike,H., 'Statistical Predictor Identification,' Annals of the Institute of Mathematical Statistics, 22, 1970, 203-217.

[6] Andrews,H.C., and Hunt,B.R., Digital Image Restoration, Prentice Hall, Englewood Cliffs, New Jersey, 1977.

[7] Bartlett,M.S., The Statistical Analysis of Spatial Pattern, Chapman and Hall, London, 1976.

[8] Beck,J. 'Perceptual groupings Produced by Line Figures,' Perception and Psychophysics, 2, 1967, 491-495.

[9] Besag,J.E., 'On the Statistical Analysis of Nearest Neighbor Systems,' Colloquia Mathematica Societatis, Janos Bolyai, Ninth European Meeting of Statisticians, Budapest, 101-105, 1972.

[10] Besag,J., 'Spatial Interaction and the Statistical Analysis of Lattice Systems (with Discussion),' Journal of the Royal Statistical Society, Series B, 36, 1974, 192-236.

[11] Beucher,S., 'Random Processes Simulations on the Texture Analyser,' Geometrical Probability, Lecture Notes in Biomathematics, No. 23, Springer Verlag, Berlin, 1978, 311-321.

[12] Billingsley,P., 'Hausdorff Dimension in Probability Theory,' Illinois Journal of Mathematics, 1960, 187-207.

[13] Billingsley,P., 'Hausdorff Dimension in Probability Theory II,' Illinois Journal of Mathematics, 1961, 291-298.

[14] Blinn,J.F., and Newell,M.E., 'Texture and Reflection in Computer Generated Images,' Communications of the ACM, 19, 542, 1976.

[15] Blinn,J.F., 'Simulation of Wrinkled Surfaces,' Computer Graphics, 12(3), 286, 1978.

[16] Bortz,A.B., Kalos,M.H., Lebowitz,J.L., and Zendejas,M.A., 'Time Evolution of a Quenched Binary Alloy: Computer Simulation of a Two-Dimensional Model System,' Physical Review B, 10, 535-541, 1974.

[17] Brodatz,P., Textures, Dover, New York, 1966.

[18] Caelli,T., and Julesz,B., 'On Perceptual Analyzers Underlying Visual Texture Discrimination,' Biological Cybernetics, 28, 167-175, 1978.

[19] Caelli,T., and Julesz,B., 'On Perceptual Analyzers Underlying Visual Texture Discrimination,' Biological Cybernetics, 29, 201-214, 1978.

[20] Catmull,E., 'Computer Display of Curved Surfaces,' Proceedings of the Conference on Computer Graphics, Pattern Recognition, Data Structures. IEEE Publication 75CH0981-1C, 11-17, 1975.

[21] Catmull,E., and Smith,A.R., '3-D Transformations of Images in Scanline Order,' Proceedings of the Siggraph Conference, Seattle, 279-285, July, 1980.

[22] Cliff,A.D., and Ord,J.K., 'Model Building and The Analysis of Spatial Pattern in Human Geography,' Journal of the Royal Statistical Society, Series B, 37, 1975, 297-348.

[23] Conners,R.W., 'Towards a set of Statistical Features which Measure Visually Perceivable Qualities of Textures,' Proceedings of the IEEE Computer Society Conference on Pattern Recognition and Image Processing, Chicago, Illinois, August, 1979, 382-390.

[24] Conners,R.W., and Harlow,C.A., 'A Theoretical Comparison of Texture Algorithms,' IEEE Transactions on Pattern Analysis and Machine Intelligence, PAMI-2, 1980, 204-222.

[25] Csuri,C., Hackathorn,R., Parent,R., Carlson,W., and Howard,M., 'Towards an Interactive High Visual Complexity Animation System,' Computer Graphics, 13, 289-299, 1979.

[26] Deguchi,K., and Morishita,I., 'Texture Characterization and Texture-based Image Partitioning using Two-Dimensional Linear Estimation Techniques,' IEEE Transactions on Information Theory, C-27, 1978, 738-745.

[27] Delp,E.J., Kashyap,R.L., Mitchell,O.R., and Abhyankar,R.B., 'Image Modelling with a Seasonal Autoregressive Time Series with Applications to Data Compression,' Proceedings of the IEEE Computer Society

Conference on Pattern Recognition and Image Processing, Chicago, Illinois, 1978, 100-104.

[28] Domb,C., 'Ising Model,' in Phase Transitions and Critical Phenomena, C. Domb and M.S. Green, editors, Academic Press, New York, 1974.

[29] Dungan,W. 'A Terrain and Cloud Computer Generation Model,' Computer Graphics, 13(2), 1979, 143-150.

[30] Ehrich,R.W., and Foith,J.P., 'A View of Texture Topology and Texture Description,' Computer Graphics and Image Processing, 8, 1978, 174-202.

[31] Feller,W., An Introduction to Probability Theory and its Applications, Volume 1, 3rd Edition, John Wiley and Sons, New York, 1968.

[32] Flinn,P.A., and McManus,G.M., 'Monte Carlo Calculation of the Order-Disorder Transformation in the Body-Centered Cubic Lattice,' Physical Review, 124, 54-59, 1961.

[33] Flinn,P.A., 'Monte Carlo Calculation of Phase Separation in a 2-Dimensional Ising System,' Journal of Statistical Physics, 10, 89-97, 1974.

[34] Fosdick,L.D., 'Calculation of Order Parameters in a Binary Alloy by the Monte Carlo Method,' Physical Review, 116, 565-573, 1959.

[35] Frisch,H.L., and Julesz,B., 'Figure-Ground Perception and Random Geometry,' Perception and Psychophysics, 1, 1966, 389-398.

[36] Froehlich,G.K., Walkup,J.F., and Asher,R.B., 'Optimal Estimation in Signal-Dependent Film-Grain Noise,' Proceedings of ICO-11 Conference, Madrid, 1978, 367-369.

[37] Galloway,M., 'Texture Analysis using Gray Level Run Lengths,' Computer Graphics and Image Processing, 4, 1974, 172-199.

[38] Galbraith, R.F. and Walley,D., 'On a Two-Dimensional Binary Process,' Journal of Applied Probability, 13, 548-557, 1976.

[39] Galbraith, R.F. and Walley,D., 'Ergodic Properties of a Two-Dimensional Binary Process,' Journal of Applied Probability, 17, 124-133, 1980.

[40] Giger,H., 'Rasterbare Texturen,' Zeitschrift fur Angewandte Mathematik und Physik, 26, 1975, 521-536.

[41] Gonzalez,R.C., and Wintz,P., Digital Image Processing, Addison-Wesley, Reading, Massachusetts, 1977.

[42] Hall,E.L., Computer Image Processing and Recognition, Academic Press, New York, 1980.

[43] Hammersley,J.M., and Handscomb,D.C., Monte Carlo Methods, Methuen and Company, London, 1964.

[44] Hammersley,J.M., 'Stochastic Models for the Distribution of Particles in Space,' Supplement to Advances in Applied Probability, 1972, 47-68.

[45] Haralick,R.M., Shanmugam,K., and Dinstein,I., 'Textural Features for Image Classification,' IEEE Transactions on Systems, Man, and Cybernetics, SMC-3, 1973, 610-621.

[46] Haralick,R.M., and Shanmugam,K., 'Computer Classification of Reservoir Sandstones,' IEEE Transactions on Geoscience Electronics, GE-11, 1973, 171-177.

[47] Haralick,R.M., 'Statistical and Structural Approaches to Texture,' Proceedings of the 4th International Joint Conference on Pattern Recognition, Kyoto, Japan, November,

1978, 45-69.

[48] Hassner,M., and Sklansky,J., 'Markov Random Field Models of Digitized Image Texture,' Proceedings of the 4th International Joint Conference on Pattern Recognition, Kyoto, Japan, November, 1978, 538-540.

[49] Hassner,M., and Sklansky,J., 'Markov Random Fields as Models of Digitized Image Texture,' Proceedings of the IEEE Conference on Pattern Recognition and Image Processing, Chicago, Illinois, 1978, 346-351.

[50] Hassner,M., and Sklansky,J., 'The Use of Markov Random Fields as Models of Texture,' University of California School of Engineering, Irvine, Irvine, California, Technical Report TP79-11, September, 1979.

[51] Hurewicz,W., and Wallman,H. Dimension Theory, Princeton University Press, Princeton, New Jersey, 1941.

[52] Isaacson,E., and Keller,H.B., Analysis of Numerical Methods, John Wiley and Sons, Inc, New York, 1966.

[53] Ising,E., Zeitscrift Physik, 31, 253, 1925.

[54] Julesz,B., 'Visual Pattern Discrimination,' IRE Transactions on Information Theory, IT-8, 1962, 84-97.

[55] Julesz,B., 'Texture and Visual Perception,' Scientific American, 212, 1965, 38-54.

[56] Julesz,B. 'Cluster Formation at Various Perceptual Levels,' in Proceedings of the International Conference on Methodologies of Pattern Recognition, Hawaii, January, 1968, S. Watanabe, Editor. Academic Press, New York, 297-315.

[57] Julesz,B., Foundations of Cyclopean Perception. University of Chicago Press, Chicago, Illinois, 1971.

[58] Julesz,B., Frisch,H.L., Gilbert,E.N., and Sheop,L.A., 'Inability of Humans to Discriminate Between Visual Textures that Agree in Second-Order Statistics - Revisited,' Perception, 2, 1973, 391-405.

[59] Julesz,B., 'Experiments in the Visual Perception of Texture,' Scientific American, 232, 1975, 34-43.

[60] Julesz,B., Gilbert,E.N., and Victor,J.D., 'Visual Discrimination of Textures with Identical Third Order Statistics,' Biological Cybernetics, 31, 137-140, 1978.

[61] Kac,M., and Ward,J.C., 'A Combinatorial Solution of the Two-Dimensional Ising Problem,' Physical Review, 88, 1332-1337, 1952.

[62] Kaizer, H., A Quantification of Textures on Aerial Photographs, Boston University Research Laboratories, Technical Note 121, 1955, AD69484.

[63] Klein,J.C., and Serra,J., 'The Texture Analyser,' Journal of Microscopy, 95, pt. 2, 1972, 349-356.

[64] Kraft,A.L., and Winnick,W.A., 'The Effect of Pattern and Texture Gradient on Slant and Shape Judgments,' Perception and Psychophysics, 2, 1967, 141-147.

[65] Lu,S.Y. and Fu,K.S., 'A Syntactic Approach to Texture Analysis,' Computer Graphics and Image Processing ,7, 1978, 303-330.

[66] McCormick,B.H., and Jayaramamurthy,S.N., 'Time Series Model for Texture Synthesis,' International Journal of Computer and Information Sciences, 3, 1974, 329-343.

[67] Mandelbrot,B.B., Fractals- Form, Chance, Dimension, W.H. Freeman and Company, San Francisco, 1977.

[68] Marr,D. 'Analyzing Natural Images: a Computational Theory of Texture Vision,' Cold Spring Harbor Symposium on Quantative Biology, 40, 647-662, 1976.

[69] Marr,D., 'Early Processing of Visual Information,' Philosophical Transactions of the Royal Society of London, Series B., 275, 483-519, 1976.

[70] Matern,B., 'Spatial Variation,' Medd. Statens Skogforskningsinstitut, Stockholm, 36(5), 1-144, 1960.

[71] Matheron,G., Elements pour une Theorie des Milieux Poreux, Masson, Paris, 1967.

[72] Matheron,G., The Theory of Regionalized Variables and its Applications, Les Cahiers du Centre de Morphologie Mathematique de Fountainbleau, 5, 1971.

[73] Metropolis,N., Rosenbluth,A.W., Rosenbluth,M.N., Teller,A.H., and Teller,E., 'Equations of State Calculations by Fast Computing Machines,' Journal of Chemical Physics, 21, 1087-1091, 1953.

[74] Mitchell,O.R., Meyers,C.R., and Boyne,W., 'A Max-Min Measure for Image Texture Analysis,' IEEE Transactions on Computers, C-26, 1977, 408-414.

[75] Modestino,J.W., and Fries,R.W., 'Stochastic Models for Images and Applications,' Pattern Recognition and Signal Processing, C.H. Chen(Ed.), Sitjhoff and Noordhoff, Alphen aan den Rijn, The Netherlands,

[76] Nevatia,R., 'Locating Object Boundaries in Textured Environments,' IEEE Transactions on Computers, C-25, 1170-1175, 1976.

[77] Newell,G.F., and Montroll,E.W., 'On the Theory of the Ising Model of Ferromagnetism,' Reviews of Modern Physics, 25, 1953, 353-389.

[78] Newman,W.M., and Sproull,R.F., Principles of Interactive Computer Graphics, Mc-Graw Hill, New York, 1979.

[79] Onsager,L., 'Crystal Statistics I: A 2-Dimensional Model with an Order-Disorder Transition,' Physical Review, 65, 117-149, 1944.

[80] Ord,J.K., and Cliff,A.D., Spatial Autocorrelation, Pion, London, 1973.

[81] Pears,A.R., Dimension Theory of General Spaces, Cambridge University Press, Cambridge, 1975.

[82] Pickard,D.K., 'A Curious Binary Lattice Process,' Journal of Applied Probability, 14, 717-731, 1977.

[83] Pratt,W.K., Faugeras,O.D., and Gagalowicz,A., 'Visual Discrimination of Stochastic Texture Fields,' IEEE Transactions on Systems, Man, and Cybernetics, SMC-8, 1978, 796-804.

[84] Ripley,B.D., 'Modelling Spatial Patterns,' Journal of the Royal Statistical Society, Series B, 8, 1977, 172-212.

[85] Rogers,A. Statistical Analysis of Spatial Dispersion, Pion Ltd., London, 1974.

[86] Rosenfeld,A., and Troy,A., 'Visual Texture Analysis,' TR 70-116, University of Maryland, June, 1970.

[87] Rosenfeld,A., and Thurston,M., 'Edge and Curve Detection for Visual Scene Analysis,' IEEE Transactions on Computers, C-20, 1971, 562-569.

[88] Rosenfeld,A., 'Some Recent Developments in Texture Analysis,' Proceedings of the IEEE Computer Society Conference on Pattern Recognition and Image Processing, Chicago, Illinois, August, 1979, 618-622.

[89] Schacter,B., Rosenfeld,A., and Davis,L.S., 'Random Mosaic Models for Textures,' IEEE Transactions on Systems, Man, and Cybernetics, SMC-8 1978, 694-702.

[90] Schacter,B., and Ahuja,N., 'Random Pattern Generation Processes,' Computer Graphics and Image Processing, 10, 1979, 95-114.

[91] Serra,J., 'Stereology and Structuring Elements,' Journal of Microscopy, 95, pt. 1, 1972, 99-103.

[92] Serra,J., and Verchery,G., 'Mathematical Morphology Applied to Fiber Composite Materials,' Film Science and Technology, 6, 1973, 141-158.

[93] Serra,J. Boolean Model and Random Sets, Centre de Morphologie Mathematique, Ecole des Mines de Paris, Report N-614, 1979.

[94] Spitzer,F., ' Markov Random Fields and Gibbs Ensembles,' American Mathematics Monthly, 78, 142-154, 1971.

[95] Sternberg,S.R., 'Parallel Architectures for Image Processing,' Proceedings of the 3rd International IEEE COMPSAC, Chicago, 1979.

[96] Sternberg,S.R., 'Language and Architecture for Parallel Image Processing,' Proceedings of the Conference on Pattern Recognition in Practice, Amsterdam, The Netherlands, May, 1980.

[97] Strauss,D.J., 'Clustering on Coloured Lattices,' Journal of Applied Probability, 14, 135-143, 1977.

[98] Tamura,H., Mori,S., and Yamawaki,T., 'Textural Features Corresponding to Visual Perception,' IEEE Transactions on Systems, Man, and Cybernetics, SMC-8, 1978, 460-473.

[99] Thompson,W.B., 'Textural Boundary Analysis,' IEEE Transactions on Computers, C-24, 1977, 272-276.

[100] Turing,A.M., 'Computing Machinery and Intelligence,' Mind, 59, 433-460, 1950.

[101] Verhagen,A.M.W., 'A 3-parameter Isotropic Distribution of Atoms and the Hard Core Lattice Gas,' Proceeedings of the Royal Society, Series A, 1977.

[102] Welberry,T.R., 'A 2-Dimensional Model of Crystal Growth Disorder,' Journal of Applied Crystallography, 6, 87-96, 1977.

[103] Welberry,T.R. and Galbraith,R.F., 'The Effect of Non-linerity on a 2-Dimensional Model of Crystal Growth Disorder,' Journal of Applied Crystallography, 8, 636-644, 1975.

[104] Welberry,T.R., 'Solution of Crystal Growth Disorder Models by Imposition of Symmmetry,' Proceedings of the Royal Society, Series A, 353, 363-376, 1977.

[105] Welberry,T.R., and Miller,G.H., 'An Approximation to a Two-Dimensional Binary Process,' Journal of Applied Probability, 14, 862-868, 1977.

[106] Weszka,J., Dyer,C. and Rosenfeld,A., 'A Comparative Study of Texture Measures for Terrain Classification,' IEEE Transactions on Systems, Man, and Cybernetics, SMC-6, 1976, 269-285.

[107] Whittle,P., 'On Stationary Processes in the Plane,' Biometrika, 41, 1954, 434-449.

[108] Yessios,C.I., 'Computer Drafting of Stones, Wood, Plant, and Ground Materials,' Computer Graphics, 13(2), 1979, 190-198.

[109] Yokoyama,R., and Haralick,R.M., 'Texture Snynthesis using a Growth Model,' Computer Graphics and Image Processing, 8, 1978, 369-381.

[110] Yokoyama,R., and Haralick,R.M., 'Texture Pattern Image Generation by Regular Markov Chain,' Pattern Recognition, 11, 1979, 225-234.

[111] Zucker,S.W., 'On the Foundations of Texture: A Transformation Approach,' TR-331, University of Maryland, September 1974.