TEXTURE CLASSIFICATION BY PATTERN KNOWLEDGE DISCOVERY

HUI SHI

TEXTURE CLASSIFICATION BY PATTERN KNOWLEDGE DISCOVERY

by

© Hui Shi

A thesis submitted to the

School of Graduate Studies

in partial fulfilment of the

requirements for the degree of

Master of Science

Department of Computer Science

Memorial University of Newfoundland

January 2007

St. John's Newfoundland

Your file Votre référence
ISBN: 978-0-494-31287-2
Our file Notre référence
ISBN: 978-0-494-31287-2

Abstract

Texture analysis has received a considerable amount of attention over the last few decades as it creates the basis of the most object recognition methods. Texture analysis mainly comprises texture classification, texture segmentation, and both of them require the important step: texture features extraction. Many approaches have been proposed either as spatial domain methods or frequency domain methods. Many texture features based on the spatial domain methods have been proposed as those methods are proven to be more superior. Texture can also be considered as a collection of patterns. Distances, directions and pixel gray-level values can determine the relationship among pixels within each pattern. Therefore, patterns are considered as the basis of textures and textures are considered to be different if they contain distinguished patterns. The procedure of pattern knowledge discovery has been started in order to find the distinctive texture patterns with gray-level deviations and distances deviations. An apriori algorithm with the joining step, cleaning step and pruning step has been introduced to find frequent patterns in order to generate higher order patterns which can be used to categorize textures. A large number of textures from the benchmark album of Brodatz have been applied and tested in the proposed method in order to prove the validity of this system and the performance is promising. The overall high accuracy shows the great encouragement from the testing procedure.

Acknowledgements

I wish to extend my appreciation to my supervisor Dr.SiWei Lu for his insightful guidance, constructive suggestions and humor. It would be impossible to finish this thesis without his help. A publication based on this thesis appears in the 36^{th} International Conference on Computers and Industrial Engineering and this paper is also under Dr.Lu's insightful guidance.

I also like to dedicate this thesis to my parents and brothers. I can always feel your love whenever I talk to you. Your encouragement backed me up during some tough times in my life.

I would like to thank the Department of Computer Science and the School of Graduate Studies for the financial support I have received during my studies.

I am also grateful to Elaine Boone who has helped me in one way or another during my studies. In addition, I am indebted to Michael Rayment on improving my spoken English and squash game skills. My appreciation also goes to Donald Craig who helped me on some programming problems.

I extend my appreciation to Dr.Rupert Paget for his help in providing Meastex database.

My appreciation also goes to Dr. John Shieh and Dr. Xue Dong Yang for their thorough examiniations on my thesis and constructive suggestions.

Last but not the least, it is impossible to enumerate the complete list of people who have offered me a hand before. I will try my best to make the best out of the experience I have gained during my stay in Memorial and Canada at large.

Contents

List of Tables

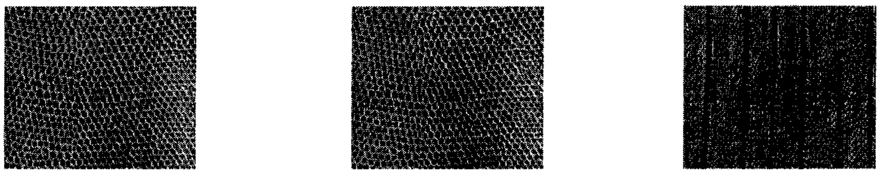
List of Figures

Chapter 1

Introduction

1.1 The Texture Classification Field

"Texture" is used to describe the surface of an object that contains a periodic organized area. People can easily observe them with their naked eyes from natural images, such as images of grasslands, leaves of trees or beach pebbles. Figure 1.1 shows some concrete examples.



Homespun woolen cloth *Lizard skin* *Woven matting*

Figure 1.1: Texture examples

More examples may include satellite multi-spectral images, electron micrographs, and microscopic images in biological or medical studies. Although it is difficult to give a universal definition of texture due to the diversity of natural and artificial textures, one could suffice by defining it as, The distinctive physical composition or structure of something, especially with respect to the size, shape,and arrangement of its parts [2]. Although no formal definition of texture exists, the three principal approaches used in image processing to describe the texture of a region are statistical, structural, and spectral [12].

Texture analysis has received a considerable amount of attention over the last few decades as it creates the basis of most object recognition methodologies. Texture analysis consists of three main parts: texture classification, texture segmentation, and texture synthesis. Texture classification refers to assigning a physical object or incident into one of a set of predefined categories. Generally speaking, the goal of texture classification is to assign an unknown sample image to one of a set of known texture classes. Therefore, it becomes obvious that a successful classification relies heavily on the efficient description of image textures. Texture segmentation, which deals with the partitioning of an image into regions that have homogeneous properties with respect to texture, and texture synthesis, the building of a model of image texture which can then be used for generating the texture. A major problem in texture analysis is that textures in the real world are often not uniform due to changes in orientation, scale or other visual appearance. In this thesis, only those textures with periodic patterns have been taken into account.

A typical texture classification experiment is usually comprised of the following steps: The first, selection of benchmark data, which may be artificial or natural, possibly obtained in a real world application. Brodatz album [4] textures are probably the most widely used image data in texture analysis literature. Other well-known data sets include VisTex [1] and MeasTex [3] textures. In this thesis, we have selected Brodatz album textures.

The second, partitioning of the image data into subimages. Texture images are often limited in terms of the number of original source images available. Therefore, in order to increase the amount of data, the images are divided into subimages, either overlapped or disjoint, of a particular window size. In this thesis, one source texture has been divided into 20 subimages randomly with the window size of 64 pixels in each row and 64 pixels in each column. The window size of subimages is determined according to the experience. If the window size is too small, it may not capture the characteristics of the whole texture and if the window size is too large, the cost of involved computation may become out of control.

The third step is the partitioning of the subimages data into training and testing sets. In order to obtain an unbiased estimate of the performance of the texture classification procedure, training and testing sets should be independent. Different approaches can be used such as the N-fold approach. This is when the collection of subimages is divided into N disjoint sets, of which the first N-1 sets serve as the training data in turn and the Nth set is used for testing. In this thesis, all subimages have been divided into two sets equally. One set is reserved for training while the other set is used for testing.

The fourth step is selection of a classification algorithm, which may involve selection of metrics or (dis) similarity measures. Selection of a classification algorithm can have a great impact on the final performance of the texture classification procedure. No classifier can succeed with poor features, but good features can be wasted with poor classifier design.

Lastly, two basic alternatives definition of the performance are available: analysis of feature values and class assignments, of which the latter is used much more often. In the former, the similarity of feature values between training and testing sets, or the separation of class clusters provided by the feature values, provides the basis for the characteristics. Tuceryan and Jain [36] divided texture analysis methods into four categories: statistical, geometrical, model-based and signal processing. An extensive literature review will be provided in the following chapters based on these four mainstream categories. From the extensive surveys on the area of texture classification, it has been concluded that those approaches on the spatial domain have been proven to be superior to other techniques. Mitchell G.A. Thomson et al [33] discovered that the role of second and third order statistics in the discriminability of natural images is crucial. Takahiro Toyoda et al [35] also proved that higher order local features play an important role in the texture analysis. However, the approaches on the extraction of meaningful texture features based on higher order statistics still remain completely unexplored. In this thesis, a new system has been created in order to classify textures from benchmark data, which are from the Brodatz Album. This new system adopts techniques in the area of knowledge discovery and uses the Apriori algorithm to discover the feature patterns, which have been used to classify textures. A new confuse table has been proposed to perform the task of classification. The main contribution of this thesis is the new approach on the classical problem and the new texture classification system, which shows a promising performance with the benchmark data on the texture classification experiment.

1.2 The Proposed System

The main objective of this system is to discover distinctive patterns inside textures and take those patterns as the descriptor of this texture in order to classify unknown data sets. This proposed system comprises the following parts. The first part is the pre-processing phase. This phase includes scanning textures into the machine and taking blocks of a texture with sufficient periodic textons as training samples and testing samples. These samples are all in the 64 x 64 uniform resolution. Texton refers to fundamental microstructures in generic natural images and basic elements in early (pre-attentive) visual perception. The second and also the most important part is the mining phase. In this part, a certain amount of elastic patterns with grey-level value intervals have been discovered through the procedure using apriori algorithm. After patterns have been discovered, these distinctive patterns will go through the third phase in order to get the five most distinctive feature patterns of each texture class. The reason of choosing five patterns is to construct a voting stragety and empirical experiments showed that too much textures may cause much computaion while too little pictures can not build a solid voting stragety. Therefore, the value is set to five. In the fourth part, training samples from the first part have been scanned in order to find the occurrence of the five most distinctive feature patterns extracted from each texture class and the result yields a confuse table for all texture class. The last part is to perform the task of classification upon testing samples. The fourth part is called the learning phase, and the last part is called the recognition phase. The purpose of the learning phase is to build a confuse table that will be used in the recognition phase. During the learning phase, if any training sample contains three out of the five most distinctive feature patterns from a certain texture class, this sample will be considered classified to this specific texture class. A predefined threshold for feature patterns is determined to help justify whether a sample contains feature patterns. If the occurrence of a feature pattern exceeds the predefined threshold in a sample, this sample is then believed to contain this feature pattern.

1.3 Structure of this thesis

This thesis is comprised of the following chapters. Chapter 2 is dedicated to the complete literature survey on the area of texture classification. More specifically, existing approaches on the feature description have been introduced and compared. Chapter 3 is for the area of classification algorithms. In Chapter 4, emphasis has been put on the area of knowledge discovery and Chapter 5 describes the implementation of the proposed system. In Chapter 6, conclusions are given and future work are suggested.

Chapter 2

Survey On The Texture Description

2.1 Introduction

An extensive literature review has been conducted on the type of texture features and the comparison among these features based on different perspectives has been analyzed and is presented in this chapter.

2.2 Texture Description

In the research area of texture classification, it is crucial to extract the features that can be used to represent the texture. A wide variety of techniques for describing image texture have been proposed in recent years. Texture analysis methods have been traditionally divided into two categories: the statistical or stochastic approach, which treats textures by using a statistical approach; and the structural approach, which is based on the concepts of texture primitives. The goal of the structural approach is to describe complex structures with simpler primitives.

Tuceryan and Jain [36]divided texture analysis techniques into four categories: statistical, geometrical, model-based, and signal processing. The four types of texture techniques are based on the surveys that follow.

2.2.1 Statistical Description

Statistical methods analyze the spatial distribution of gray values through computing local features at each point in an image. The term, statistical feature, means a set of statistics from the distribution of the local features. Depending on the number of pixels, the local feature statistical methods can be classified into first-order (one pixel), second-order (two pixels) and higher-order (three or more pixels) statistics. The most widely used statistical methods are co-occurrence features [23] and gray level differences [40]. The latter method has inspired many variations. Other statistical approaches include the autocorrelation function [16], which has been used for analyzing the regularity and coarseness of a texture, and the approach of gray level run lengths [10].

2.2.1.1 Descriptions of the Gray-Level Histogram

A histogram deals with the distribution of pixels' value in an image and consists of two approaches: the global histogram and the local histogram. This process can be denoted by the expression g(x,y)=T[f(x,y)] where f(x,y) is the input image, g(x,y) is the processed image and T is an operator on f, as defined over some neighborhood of (x,y). The essential difference between these two approaches is the selection of the neighborhood over a point (x, y). If the neighborhood is a single pixel, the processed image g will depend only on the value of f at the pixel (x, y) and T will become a gray-level transformation function. Otherwise, T will become a local histogram function.

Statistical moments of the gray-level histogram of an image or a region of an image are one of the simplest approaches for texture description. Let z be a random pixel denoting gray level and let p(zi) be the corresponding normalized histogram where i = $0,1,2,\cdots$, L-1 where L is the number of distinct gray levels.

The moment is defined by the order of n. The second moment is especially important in the texture description because a measure of gray-level contrast can be used to form descriptors of smoothness. The value of the second moment is 0 when the areas are of constant intensity, and 1 when the areas are of the maximum contrast. Therefore, the larger the value of the second moment is, the more obvious the contrast will be. The third moment is defined in order to measure the skewness of the histogram of the image. If the result of the third moment is of negative sign, this means the degree of the symmetry of the histogram is to the left. Otherwise, it is to the right. This measure gives an idea as to whether the gray levels are toward the dark side or light side of the mean value.

Aside from the nth moment approaches, other texture measures, which are based on histograms, are available. This measure describes the smoothness of an image. The smoother the image is, the larger this measure will be. It is always used together with the second moment measure. Another measure is called the average entropy measure. The measure of variability is entropy and the value is 0 when the image is constant. The coarser the image is, the larger the value of entropy.

2.2.1.2 Co-occurrence Matrix Features

Measures of texture using histograms lack the information of the relative position of pixels. Therefore, it is natural to solve this problem by taking the information of the positions of pixels with equal or nearly equal gray-level values into account. A condition of the probability density function is introduced which is defined as P(i,j|d,q). Each P denotes a probability of concurrence of a pair of gray levels (i, j) at a given displacement operator with a distance d and an angle q. Therefore, the estimated values are an estimation of the joint probability that a pair of points match the given displacement operator and all those estimation values can generate a matrix, which is called a gray-level co-occurrence matrix. According to the different distances and angles, a number of co-occurrence matrices can be created. Haralick [23]has proposed the benchmark properties of textures such as energy, contrast, correlation, entropy and local homogeneity based on the co-occurrence matrix. These features are basically varieties of characterization of the contents of the co-occurrence matrices.

2.2.1.3 Gray Level Differences Features

The gray-level differences method is similar to the co-occurrence approach. The contrast between the two is that the difference of the gray levels of the pair of pixels is utilized in the gray-level differences method while only absolute gray levels are used in the co-occurrence matrices approach. It is possible to achieve invariance against changes in the overall luminance of an image through the gray level differences approach. Another advantage is that the differences tend to have a smaller variability to the absolute gray levels in natural textures, thus resulting in more compact distributions. Weszka [40] proposed the use of the mean difference, the entropy of differences, a contrast measure, and an angular second moment as gray level difference features.

2.2.1.4 Autocorrelation Function

The autocorrelation function holds the characteristics of indicating the sizes of primitives because the function will drop off and rise according to the size of primitives. If the size of a primitive is relatively large, its autocorrelation function value may decrease slowly. Otherwise, its autocorrelation function value may decrease rapidly. The spatial information can therefore be characterized by the correlation coefficients.

2.2.1.5 Gray Level Run Length

Gray level run length is a measure that is used to evaluate the degree of coarseness of textures. A gray level run is a set of neighboring pixels of the same gray level value. A matrix P can be made where each element P(i,j) denotes the number of runs with the length j for the gray value i. Therefore, for a coarse texture, long runs should appear quite often while short runs should occur more frequently for fine textures. It is easy to notice that this approach is sensitive to noise.

2.2.1.6 Brief Summary of Statistical Features

As discussed before, the histogram is one of the simplest approaches in the texture description. It is simple because it is derived from the distribution of the image histogram. It does not contain the relative information among pixels, however, it is widely adopted due to its easy implementation.

Features using the co-occurrence matrix have proven to be superior features by many researchers [40, 20]since they characterize the spatial relationships of gray levels in an image. However, the co-occurrence approach may tend to consume a large amount of computation because many matrices have to be computed. In addition, there is a lack of necessary guidance in choosing features that are derived from the co-occurrence matrices. The autocorrelation function is impervious to noise but is extremely expensive computationally. The Gray Level Run Length is very sensitive to noise, which does not make it the optimal method for gray value images. It is nevertheless suitable for binary images [28].

2.2.2 Geometrical Description

Geometrical methods consider texture to be composed of texture elements or primitives. The analysis method usually depends on the geometric properties of texture primitives. Once the primitives are determined in a textured image, there are two major approaches to analyze the texture. One is to compute the statistical properties from extracted texture elements and utilize these properties as texture features. The other approach is to extract the placement rule that describes the texture and take these rules as texture features.

Image edges are the most often used texture primitives and one of the many existing approaches is the generalized co-occurrence matrix which describes the second-order statistics of edges. The generalized co-occurrence matrix considers an arrangement of gray values of a pixel neighborhood as a primitive. Here is an example: suppose a neighborhood with the size 3 3 with 4 gray levels for each pixel will result in 49 different primitives. Histogram statistics will be used to indicate the frequency of the occurrence of all the primitives in the image and thus reveal the texture information. Obviously, the problem with this method is the heavy computation due to the high dimensionality for the probability distribution.

Another approach is to take into account the arrangement of texture primitives. The primitives can be as simple as a gray value, but they usually are a collection of pixels. Fu [9]proposed this approach to define the placement rule using a grammar tree. Therefore, a texture is considered to be a string defined by the grammar whose terminals are those primitives. Zucker [44] also proposed a model based on a similar idea.

2.2.3 Model-based Description

This approach supposes texture as a realization of a stochastic process which can be determined by a set of parameters. These parameters will then be used to describe this texture in classification and segmentation via different operations because they captured the essential perceived qualities of textures. Models generally can be divided into two categories: pixel-based model and region-based model. Pixel-based models consider an image as a collection of pixels where region-based models regard an image as a set of patterns placed according to given rules. The pixel-based models assume no spatial interaction between neighboring pixels and the image to be processed is assumed as the sum of a deterministic polynomial and additive noise [13], while the region-based models take the interaction among neighboring pixels into account. Generally speaking, the basic idea underlying model-based approaches is that the intensity function, which is an image, is considered to be a combination of a function representing the known structural information on the image surface and an additive random noise sequence. Many researchers have been interested in the model-based approaches. The most commonly used models are the Markov Random Field (MRF), the Gibbs Random Field (GRF) and the 2-D Autoregressive (AR).

Random field models analyze spatial variations in two dimensions and they consist of two models at large. The Global random field model treats the entire image as a realization of a random field while the local random field model assumes relationships of intensities in small neighborhoods.

Markov Random Fields (MRF) are multidimensional generalizations of Markov Chains, which are defined in terms of conditional probabilities based on spatial neighborhoods (so called Markov neighbors) [42, 27]. There are different orders of a neighborhood and each neighborhood corresponds to a clique, which is a graph whose vertex set is composed of vertices such that each one is a neighbor of all others. Those parameters associated with the cliques of a given neighborhood configuration determine a Markov Random Field or MRF and these parameters are used to form a feature space in order to be used for the mission of texture classification and segmentation. The same idea is applied to the Gibbs Random Field (GRF) and the only difference is that GRF is a global random field model while MRF is a local random field model.

The Autoregressive Model considers an image as a linear combination of the neighboring values with random noise values. The coefficients of these combinations could be considered as a set of features which explicitly express the spatial relation of each pixel with its neighbors. Both the Random Field model (RF) and the Autoregressive model (AR) consider the image as a linear combination of a set of parameters and it has been pointed out that the estimation of the parameters in the AR model is less difficult than that of the Random Field Models [25, 30].

In order to use model-based approaches to describe texture images, there are generally three problems that need to be considered: (1) the selection of appropriate models and its order; (2) estimation of parameters for the chosen model; and (3) selecting appropriate classification techniques. Among these problems, the first two are interrelated in that the chosen model and its order determine the estimation of the parameters. A model with a higher order neighborhood increases the accuracy of the chosen model, however, makes the estimation job much harder to accomplish.

Practically, it is not easy to choose an appropriate neighborhood order due to the heavy computation involved in this procedure. Therefore, a neighborhood is empirically chosen with a fixed size.

For the estimation part of model-based approaches, Least Square Estimation (LSE) and Maximum Likelihood Estimation (MLE) are commonly adopted methods for estimation. Research has shown that the results obtained by LSE and by MLE are nearly the same. However, the former is easier for computation. Therefore, Least Square Estimation is frequently used in the area of parameter estimation.

2.2.4 Signal Processing Description

Methods that fall into this category analyze the frequency content of the image. The basic idea underlying this approach is that the transferred image from the spatial domain into the frequency domain may reveal useful information that is hard to discover in the spatial domain, but is easily spotted in the frequency domain. One of the approaches is called the Fourier Transformation, which was proposed by the French mathematician Fourier in 1807. The philosophy behind this transformation is that any function can be expressed as the integral of sines and/or cosines multiplied by a weighing function and the transferred function can be recovered completely via an inverse process, with no loss of information. Because an image with the size of MxN can be regarded as a two-variable discrete function, these two equations can be easily extended to the two-dimensional discrete Fourier transformation.

The most important of texture features that can be reflected in the frequency domain is that fine textures are rich in high frequencies and course textures tend to dominate in the low frequencies. A good analogy is to compare the Fourier transform to a glass prism [11]. As a prism has the capabilities to separate light into various color components, which is on its wavelength content, the Fourier transform can be viewed as a "mathematical" prism that separates a function into various components based on frequency content. This is the key concept that lies behind the Fourier transform. Numerous papers have been published in terms of this discovery in the area of texture analysis and the common techniques are trying to catch those components separated by the Fourier transform and use them to represent textures. From the perspective of functions, those components can be viewed as the coefficient values in the transformed functions. There are still other transforms available besides the Fourier transform, such as the Hadamand and Slant transformation. However, it is reported that no big difference has been found among those methods [43].

Although the Fourier transform has been popular for quite a long time, it still has drawbacks which give room for the introduction of the wavelet transform. Unlike the Fourier transform that only reveals frequencies, the Wavelet transform reveals more than those contents in terms of frequency. Because the Wavelet transform is based on small waves, which are called wavelets, of varying frequency and limited duration, this characteristic enables the Wavelet transform to provide the equivalent of a musical score for an image. In another way, it provides not only what notes (or frequencies) to play, but also when to play them. Typical Fourier transforms lost temporal information in the process of transformation.

A Wavelet transform basically incorporates and unifies techniques subband coding from signal processing and pyramidal image processing. The philosophy behind this technique is that features of images (or signals) can be revealed at more than one resolution, and thus those features that might go undetected at one resolution may become clear at another resolution. It was proposed by Burt and Adelson that an image pyramid is a collection of decreasing resolution images arranged in the shape of a pyramid. The bottom of the pyramid contains the highest resolution of the image being processed and the top is the lowest-resolution approximation of the image.

An original image can be regarded as a collection of images with decreasing-resolution approximations and residuals. For instance, the original image, which is level j, can also be viewed as the approximation at level j-1 plus the prediction residual of level j. Level j prediction residual comes from the difference between the level j approximation and the level j-1 approximation. Therefore, in the area of multi-resolution analysis (MRA), a scaling function ψ(x) is used to create a series of approximations of a function or an image and a wavelet function ω(x) is used to encode the differences in information between adjacent approximations, also known as detail coefficients where $c_{j_0}(k)$ are called the approximation coefficients and $d_j(k)$ are referred to as the detail. Wavelet transforms share the same characteristic with

Fourier transforms in that any function can be expressed as a lineal combination of expansion functions. However, in the Wavelet transform, approximation coefficients and the detail are the goals. Suppose an image with the size of m and n, $W_\psi(j_0, m, n)$ is the approximation coefficients; W_ψ^H,W_ψ^V,W_ψ^D stands for the horizon detail, the vertical detail and the diagonal detail for scales j when j is greater or equal to.

A Wavelets transform is achieved by using a window function, (approximation filters), whose width changes as the frequency changes. If the filter is a Gaussian function, the obtained transform is called the Gabor transform [21]. Texture description under the wavelet transform can be found by filtering the image with a bank of filters. Texture features are then extracted from the filtered images. A well-known family of wavelets called Cohen-Daubechies-Feauveau wavelets is used widely as the decomposition filters (or approximation filters). Another commonly adopted technique is called the wavelet packets [6]. Basically, wavelet packets are nothing more than conventional wavelet transforms in which the details are iteratively filtered.

2.2.5 Relationship Among Approaches

Commonly used texture features have been introduced. It is impractical to track down all existing approaches in this area due to the large quantities of papers published in the last thirty years. Previous introductions, however, have included the mainstream approaches in the texture analysis area and interlinks [34] among approaches have been shown in the following figure 2.1. The figure 2.1 represents interlinks among approaches in the square. It shows that different approaches may have links among themselves.

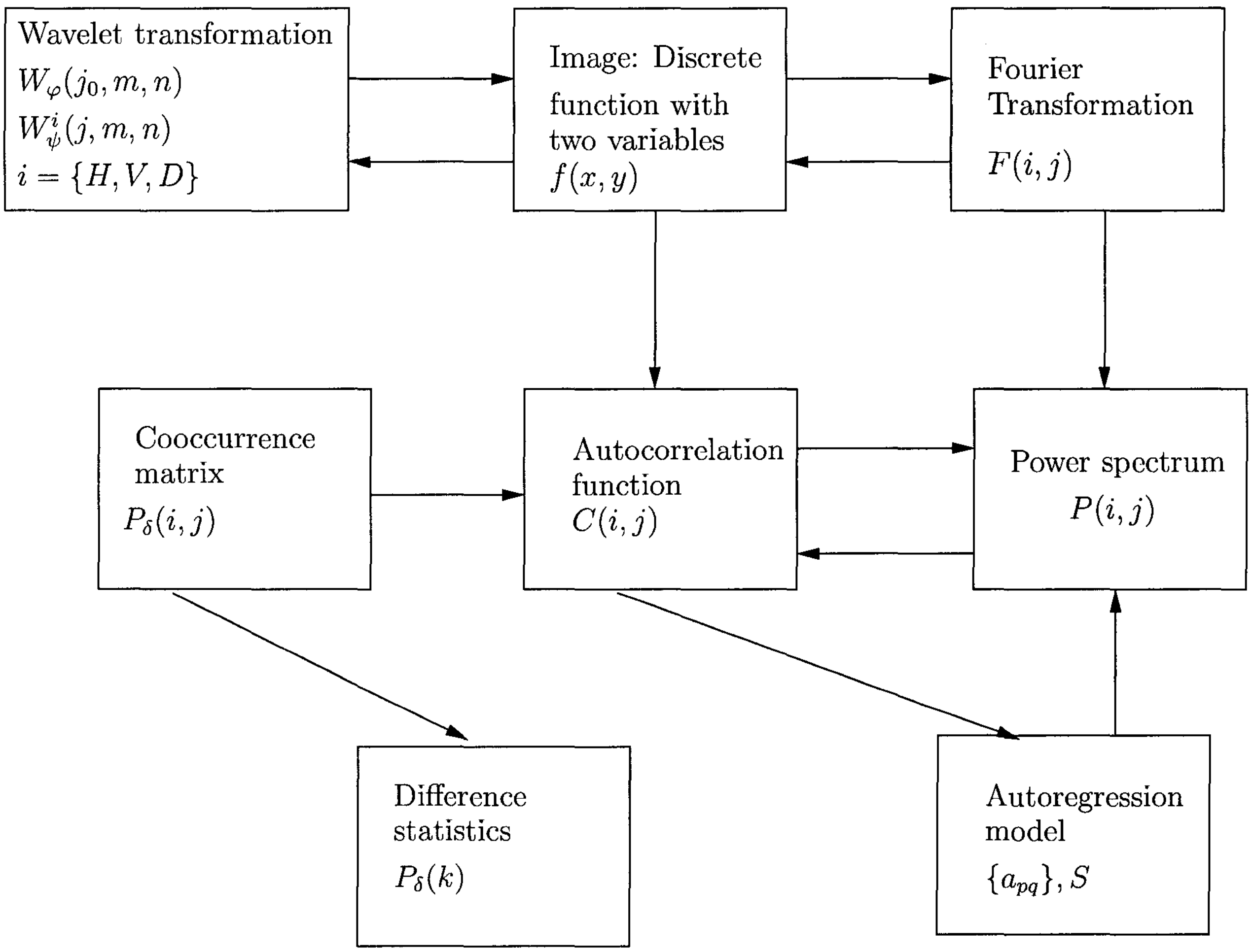


Figure 2.1: Interlinks among texture analysis approaches

2.3 Conclusion

In this section, texture features have been reviewed and their relationship has been revealed also. This section can serve as a fundamental cornerstone for understanding texture features as the most important techniques and concepts have been explained here.

Chapter 3

Common Classification Techniques

3.1 Introduction

After the texture description has been given, the next step is to find a suitable classification algorithm. Among the most widely used are parametric statistical classifiers derived from the Bayesian decision theory; nonparametric approaches like k-nearest neighbor classifier; the decision tree induction approach and the artificial neural network approach. In this section, techniques on decision tree induction, Bayesian classifier, artificial neural network and kNN classifiers have been briefly introduced.

3.1.1 Decision Tree Induction

Decision Tree Induction is a common technique used widely for the purpose of classification. A decision tree is a chart-like tree structure where each internal node denotes a test on an attribute, each branch represents an outcome of the test, and each leaf node represents a class or class distribution. Decision Tree learning is a supervised learning method and it is best suited for problems such as instances that are represented by attribute-value pairs; the target function having discrete output values. The Decision Tree approach classifies instances by sorting them down the tree from the root node to the leaf nodes. Each non-leaf node is connected to a test that splits its set of possibilities to subsets in terms of different test results. And each branch indicates the direction according to the result of a particular test and the leaf node is connected to a set of possible answers. When the classification tree has been constructed, the classification rule is easy to build. Therefore, the decision tree induction consists of the following workflow represented in Figure 3.1:

$$Data \longrightarrow DecisionTree \longrightarrow DecisionRules \longrightarrow Classification\ or\ Prediction$$

Figure 3.1: Procedure of Decision Tree Induction

The step of constructing a decision tree from a dataset is one of the most important components. The key step during the construction is to determine the order of attributes, which is useful to build small trees. In order to select attributes, the concept of average entropy is introduced. Entropy is a measure from information theory; it characterizes the degree of purity or homogeneity for a collection of samples. The approach is to find the average entropy of each attribute and choose the attribute with the minimum value of average entropy. In order to calculate the average entropy, it is necessary to know the concept of n_b, n_{bc} and n_t where n_b is the number of instances in branch b, n_{bc} is the number of instances in branch b of class c and n_t is the total number of instances in all branches. Equation 3.1 is the formula to calculate the average entropy.

$$\sum_b[(\frac{n_b}{n_t}) \times (\sum_c -(\frac{n_{bc}}{n_b})log_2(\frac{n_{bc}}{n_b}))] \tag{3.1}$$

The attribute with the minimum value of the average entropy has been selected and the decision tree is therefore constructed. The decision tree should go through a pruning algorithm in order to improve accuracy by removing tree branches. After this step, decision trees can easily be converted to classification IF-THEN rules in order to suit the task of classification.

3.1.1.1 Bayesian Classifier

Bayesian classifiers are statistical classifiers and are based on Bayes theorem. Bayes theorem is briefly introduced as follows: Let X be a data sample whose class label is unknown. Let H be some hypothesis, such as the data sample X belongs to a specified class C. For classification problem, the value of $P(H|X)$ determines the probability that the hypothesis H holds given the observed data sample X. Bayes theorem indicates that $P(H|X)$ can be determined through $P(X|H)$,$P(H)$ and $P(X)$ in the Equation 3.2

$$P(H|X) = \frac{P(X|H)P(H)}{P(X)} \tag{3.2}$$

Where P(X|H) is the posterior probability of X conditioned on H; P(H) is prior probability of H and P(X) is the prior probability of X.

There are two approaches in Bayesian classification: one is called the naive Bayesian classifier and the other one is the Bayesian Belief Network. Naive Bayesian Classifier assumes that the effect of an attribute value on a given class is independent of the values of the other attributes, which means there are no dependence relationships among attributes. The Bayesian Belief Networks specify joint conditional probability distributions. They allow class conditional independencies to be defined between subsets of variables. Thorough presentations of Bayesian classification can be found in textbooks of [8, 37, 33].

3.1.1.2 Artificial Neural Network

A neural network is a set of connected input/output units where each connection has a weight associated with it. During the learning phase, the network learns by adjusting the weights so as to be able to predict the correct class label of the input samples. Neural networks involve long training times and are therefore more suitable for applications where this is feasible. They require a number of parameters that are typically best determined empirically, such as the network topology. Neural networks have been criticized for their poor interpretability since it is difficult for humans to interpret the symbolic meaning behind the learned weights. Advantages of neural networks, however, include their high tolerance to noisy data as well as their ability to classify patterns on which they have not been trained. An extensive survey of applications of neural networks in industry, business, and science is provided in [41].

3.1.1.3 K-nearest Neighbor Classifier

Nearest neighbor classifiers are based on learning by analogies. The training samples are described by n-dimensional numeric attributes. Each sample represents a point in an n-dimensional space. In this way, all of the training samples are stored in an n-dimensional pattern space. When given an unknown sample, a k-nearest neighbor classifier searches the pattern space for the k training samples that are closest to the unknown sample. The k nearest neighbors of the unknown sample refers to these k training samples. The Euclidean of distance is defined to measure the closeness between two points. The unknown sample is assigned the most common class among its k nearest neighbors. When k=1, the unknown sample is assigned the class of the training sample that is closest to it in pattern space. Nearest neighbor methods are discussed in many statistical texts on classification, such as [8, 15]. Additional information can be found in [32, 18].

3.2 Conclusion

In this part, a brief introduction has been given to classification algorithms. There are numerous comparisons on the different classification methods, and the comparison remains a research topic. No single method has been found to be superior over all others for all data sets. Empirical comparisons [7, 5, 31] show that the accuracies of many algorithms are sufficiently similar and their differences are statistically insignificant, while training times may differ substantially. In general, most neural network and statistical classification methods involving splines tend to be more computationally intensive than most decision tree methods.

Chapter 4

Knowledge Discovery

4.1 Introduction

A review on the area of knowledge discovery has been conducted in this chapter. Knowledge Discovery in Database (KDD) is concerned with extracting useful information from databases. Because gray level images can be viewed as two-dimensional matrices, these matrices can be considered as large databases where useful information can be discovered by Knowledge Discovery techniques. A typical Knowledge Discovery System (KDS) generally includes three components: data warehouse, data mining and postprocessing. In this chapter, these three components have been introduced and emphasis has been put on the part of data mining procedure.

4.2 Data Warehouse

A data warehouse is a subject-oriented, integrated, time-variant, and nonvolatile collection of data organized in support of management decision-making [14]. It is easy to understand what a data warehouse is by comparing data warehouse with regular relational database systems, which most people are familiar with. The major task of a relational database system is to perform on-line transaction and query processing (OLTP). They cover most of the day-to-day operations of an organization, such as purchasing and inventory. Data warehouse systems, on the other hand, serve users or knowledge workers in the role of data analysis and decision-making. These systems are known as on-line analytical processing (OLAP) systems. In order to build a data warehouse, data preprocessing is an essential issue, as real-world data tend to be incomplete, noisy, and inconsistent. Data preprocessing includes data cleaning, data integration, data transformation, and data reduction. Detailed discussion can be found in [26, 38, 22].

4.3 Data Mining

The types of knowledge to be mined specify functions of data mining and these knowledge types include concept description, association, classification, clustering, or evolution analysis. Among these functions, an association rule mining searches for interesting association or correlation relationships within a large set of data items and therefore receives much attention.

4.4 Association Rule Mining

The association rule [24] is very useful in the market basket analysis because the results may be used to plan marketing or advertising strategies, as well as catalog design. Let $i = \{i_1, i_2, \cdots, i_m\}$ be a set of items. Let D be a set of database transactions where each transaction T is a set of items such that $T \in \iota$. Let A be a set of items. A transaction T is said to contain A if and only if A$\in$T. An association rule is an implication of the form A$\Rightarrow$B, where A$\subset \iota$, B$\subset \iota$, and A$\bigcap B = \emptyset$. The support s of the rule A$\Rightarrow$B means the percentage of transactions in D that contain A$\bigcup$B. This is taken to be the probability, P(A$\bigcup$B). The confidence c of the rule A$\Rightarrow$B means the percentage of transactions in D containing A that also contains B. This is taken to the conditional probability, P(B|A). Association rule mining is a two-step process: the first step is to find all frequent itemsets that are defined in terms of the predefined value of support s; the second step is to generate associate rules from the frequent itemsets and these rules must satisfy predefined values of support s and confidence c.

4.4.1 Apriori Algorithm

Apriori is an influential algorithm for mining frequent itemsets. This algorithm uses prior knowledge of frequent itemset properties. Apriori employs an iterative level-wise search, where k-itemsets are used to explore (k+1)-itemsets. First, the set of frequent 1-itemsets is found. This set is denoted as L_1. L_1 is then used to find L_2, the set of frequent 2-itemsets, which is used to find L_3, and so on, until no more frequent k-itemsets can be found. Therefore, the finding of each L_k requires one full scan of the database. As the searching space for frequent itemsets may become huge, it is necessary to introduce the Apriori property in order to improve the efficiency of the level-wise search. The Apriori property is described as follows: all nonempty subsets of a frequent itemset must also be frequent. Given an itemset I, if I does not satisfy the minimum support threshold min_sup, then I is considered as not frequent, that is P(I)<min_sup. As any superset of I cannot become more frequent than I, superset of I is not frequent either. Therefore, the searching space is reduced.

Once the frequent itemsets are mined from transactions, it is straightforward to generate strong association rules from them. Strong association rules should satisfy both minimum support and minimum confidence. Therefore, for each frequent itemset L, generate all their nonempty subsets S; output the rule S⇒(L-S) if its confidence is equal or greater than the predefined minimum confidence threshold.

4.5 Post processing

The step of post processing refers to how the data mining system displays the discovered patterns. The visualization of discovered patterns in various forms can help users with different backgrounds to identify patterns of interest and to interact or guide the system in further discovery. Many good examples and visual snapshots can be found in [39, 17].

4.6 Conclusion

In this chapter, a brief introduction has been given to the three components of the data mining system. Different data mining techniques may be adopted according to the types of the knowledge to be mined. Emphasis is put on the Association rule mining in that it may reveal the hidden relationship among itemsets within the large databases or data warehouses. And this feature may be applied to the image processing area for finding hidden patterns. Apriori algorithm is widely used to find association rules and its variations including hashing and transaction reduction can be used to make the procedure more efficient. Other variations include partitioning the data, and sampling the data. These variations can reduce the number of data scans required.

Chapter 5

The Proposed System

5.1 Introduction

It is natural for human beings to find the distinctiveness within texture images because human eyes can detect those patterns that occur frequently and take those patterns to differentiate texture images. Similar to the way that human beings detect those features, our system has grasped the uniqueness of textures and uses this uniqueness to classify the unknown features.

In this chapter, the construction of a new system has been explained in order to discover those peculiar texture patterns which may represent textures. Those patterns are discovered through a data mining procedure. The Apriori algorithm has been adopted in order to find those frequent patterns.

5.2 Distinctive Texture Patterns

Consider a M x N pixel size of textured image I(m,n) where m is the number of rows and n is the number of columns. The sample containing enough regular patterns is obtained from a textured image; practically, a sample with the size of 64x64 is adopted. A knowledge discovery process has been applied in order to find distinctive patterns inside the sample. Here, the distinctive patterns refer to patterns that are contained in this textured sample only. The number of pixels with gray level i is denoted as the frequency n_i, where $0 \leq i \leq k$ and k is the maximum gray level. Table 5.1 shows all one-pixel patterns in a sample, which is called the candidate one-pixel itemset C_1.

Table 5.1: Candidate one-pixel itemset C_1

Frequency	n_0	n_1	n_2	$\cdots$	n_i	n_{k-1}	n_k
Itemset C_1	$\{g_0\}$	$\{g_1\}$	$\{g_2\}$	$\cdots$	$\{g_i\}$	$\{g_{k-1}\}$	$\{g_k\}$

Table 5.2 contains the gray levels with frequency exceeding a predefined threshold $t_1, g_{k_p}, g_{k_q} \in L_1$, if $n_{k_p}, n_{k_q} > t_1, 0 \leq k_p, k_q \leq k$ and $p, q \leq \|L_1\|$ After the frequent

Table 5.2: Candidate one-pixel itemset L_1

Frequency	n_{k_1}	n_{k_2}	n_{k_3}	$\cdots$	n_{k_p}	n_{k_q}	$\cdots$
Itemset L_1	$\{g_{k_1}\}$	$\{g_{k_2}\}$	$\{g_{k_3}\}$	$\cdots$	$\{g_{k_p}\}$	$\{g_{k_q}\}$	$\cdots$

one-pixel itemset has been discovered, we start to find patterns with two frequent one-pixel itemsets in the itemset L_1. Figure 5.1 shows four types of patterns with two pixels.

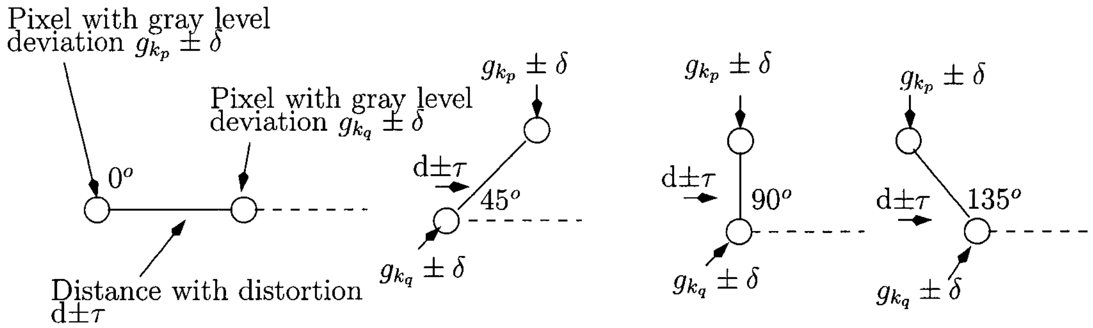


Figure 5.1: Four types two-pixel patterns with deviations

In Figure 5.1, τ stands for the distance deviation; δ for the gray-level deviation; θ represents one of four degrees: 0^o,45^o,90^o and 135^o; the value of τ and δ are determined according to correspinding textures. The concepts of deviations are necessary because textures in reality are subject to the noise. Deviations may help find the essential patterns without being disturbed by noise.

The candidate two-pixel itemset C_2 can be discovered from frequent one-pixel itemset L_1. Table 5.3 shows the candidate two-pixel itemset C_2. Each entry in C_2 is represented as $\{g_{k_p}, g_{k_q}, \theta_i, d_j\}$ in order to reflect the information of angle and distance for patterns. Here, g_{k_p} and g_{k_q} are from frequent one-pixel itemset L_1.

Table 5.3: Candidate two-pixel itemset C_2

$\{g_{k_1}, g_{k_1}, 0^o, d_3\}$	$\{g_{k_1}, g_{k_1}, 45^o, d_3\}$	$\{g_{k_1}, g_{k_1}, 90^o, d_3\}$	$\{g_{k_1}, g_{k_1}, 135^o, d_3\}$
$\{g_{k_1}, g_{k_1}, 0^o, d_4\}$	$\{g_{k_1}, g_{k_1}, 45^o, d_4\}$	$\{g_{k_1}, g_{k_1}, 90^o, d_4\}$	$\{g_{k_1}, g_{k_1}, 135^o, d_4\}$
......	......	......	......
$\{g_{k_1}, g_{k_1}, 0^o, d_9\}$	$\{g_{k_1}, g_{k_1}, 45^o, d_9\}$	$\{g_{k_1}, g_{k_1}, 90^o, d_9\}$	$\{g_{k_1}, g_{k_1}, 135^o, d_9\}$
$\{g_{k_1}, g_{k_2}, 0^o, d_3\}$	$\{g_{k_1}, g_{k_2}, 45^o, d_3\}$	$\{g_{k_1}, g_{k_2}, 90^o, d_3\}$	$\{g_{k_1}, g_{k_2}, 135^o, d_3\}$
......	......	......	......
$\{g_{k_1}, g_{k_2}, 0^o, d_9\}$	$\{g_{k_1}, g_{k_2}, 45^o, d_9\}$	$\{g_{k_1}, g_{k_2}, 90^o, d_9\}$	$\{g_{k_1}, g_{k_2}, 135^o, d_9\}$
......	......	......	......
$\{g_{k_p}, g_{k_p}, 0^o, d_3\}$	$\{g_{k_p}, g_{k_p}, 45^o, d_3\}$	$\{g_{k_p}, g_{k_p}, 90^o, d_3\}$	$\{g_{k_p}, g_{k_p}, 135^o, d_3\}$
$\{g_{k_q}, g_{k_q}, 0^o, d_9\}$	$\{g_{k_q}, g_{k_q}, 45^o, d_9\}$	$\{g_{k_q}, g_{k_q}, 90^o, d_9\}$	$\{g_{k_q}, g_{k_q}, 135^o, d_9\}$

After the candidate two-pixel itemset C_2 has been constructed, frequencies of these two-pixel patterns are calculated by the following equations:

$$f_{0^o,d}(g_{k_p}, g_{k_q}) = |\{[(k,l),(m,n)] \in D : k-m=0, |l-n| = d \pm \tau, I(k,l) = g_{k_p} \pm \delta \\ , I(m,n) = g_{k_q} \pm \delta\}| \tag{5.1}$$

$$f_{45^o,d}(g_{k_p}, g_{k_q}) = |\{[(k,l),(m,n)] \in D : (k-m = d \pm \tau, |l-n| = -(d \pm \tau)) \\ OR\ (k-m = -(d \pm \tau), |l-n| = d \pm \tau) I(k,l) = g_{k_p} \pm \delta, I(m,n) = g_{k_q} \pm \delta\}| \tag{5.2}$$

$$f_{90^o,d}(g_{k_p}, g_{k_q}) = |\{[(k,l),(m,n)] \in D : |k-m| = d \pm \tau, |l-n| = 0, I(k,l) = g_{k_p} \pm \delta \\ , I(m,n) = g_{k_q} \pm \delta\}|| \tag{5.3}$$

$$f_{135^o,d}(g_{k_p}, g_{k_q}) = |\{[(k,l),(m,n)] \in D : (k-m = d \pm \tau, l-n = d \pm \tau) \\ OR\ (k-m = -(d \pm \tau), l-n = -(d \pm \tau)) I(k,l) = g_{k_p} \pm \delta, I(m,n) = g_{k_q} \pm \delta\}| \tag{5.4}$$

Equation 5.1 is used to find the occurrence of two pixels within a certain distance and the angle between them is 0 degrees;Equation 5.2 is to find the occurrence of two pixels within a certain distance and the angle between them is 45 degrees;Equation 5.3 is to find the occurrence of two pixels within a certain distance and the angle between them is 90 degrees and Equation 5.4 is to find the occurrence of two pixels within a certain distance and the angle between them is 135 degrees.

Two-pixel patterns whose frequency exceeds the predefined threshold t_2 are kept in the frequent two-pixel itemset L_2. This step is called pruning. Frequent two-pixel patterns are used to discover frequent three-pixel patterns according to the Apriori algorithm. During this process two types of two-pixel patterns have to be removed. The first type is patterns that belong to the background, which should be removed; the other type with the equal θ and similar gray levels and distances should be removed in order to reduce redundancy. Table 5.4 contains the frequent two-pixel itemset L_2. Each entry in L_2 is represented as $\{g_{p_x}, g_{p_y}, \theta_i, d_j\}$, where $g_{p_x}, g_{p_y} \in \{g_{k_p}\}$. After the pruning, frequent two-pixel patterns are linked into three-pixel patterns

Table 5.4: Frequent two-pixel itemset L_2

$\{g_{p_1}, g_{p_2}, 0_o, d_3\}$	$\{g_{p_1}, g_{p_2}, 45_o, d_3\}$	$\{g_{p_1}, g_{p_2}, 90_o, d_3\}$	$\{g_{p_1}, g_{p_2}, 135_o, d_3\}$
$\{g_{p_1}, g_{p_3}, 0_o, d_3\}$	$\{g_{p_1}, g_{p_3}, 45_o, d_3\}$	$\{g_{p_1}, g_{p_3}, 90_o, d_3\}$	$\{g_{p_1}, g_{p_3}, 135_o, d_3\}$
......	......	......	......
$\{g_{p_x}, g_{p_y}, 0_o, d_9\}$	$\{g_{p_x}, g_{p_y}, 45_o, d_9\}$	$\{g_{p_x}, g_{p_y}, 90_o, d_9\}$	$\{g_{p_x}, g_{p_y}, 135_o, d_9\}$

to form the candidate three-pixel itemset C_3. In Figure 5.2, an elastic three-pixel pattern is illustrated. Let L^m_{k-1},L^n_{k-1} be itemsets in L_{k-1}. The notation $L^m_{k-1}[j]$

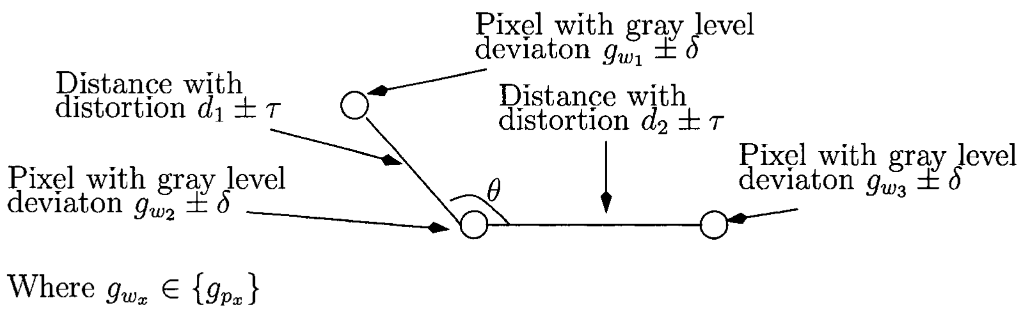


Figure 5.2: Elastic three-pixel pattern with gray level deviation

refers to the jth pixel in L^m_{k-1}. The join operation $L_{k-1} \times L_{k-1}$ is performed, where two members of L_{k-1} are linkable if one pixel gray level has the same gray value deviation as the other one. That is, members L^m_{k-1} and L^n_{k-1} of L_{k-1} are structurally linkable if $L^m_{k-1}[i], L^n_{k-1}[j] \in [g-\delta, g+\delta]; i,j \in [l, k-l], m \neq n$. An example in Figure 5.3 is used to explain how two two-pixel patterns have been linked to become a three-pixel pattern. In the Figure 5.3, $L^1_2[2], L^2_2[1] \in [g_2-\delta, g_2+\delta]$. Therefore, L^1_2 and L^2_2 are linked together to form a three-pixel pattern. Entry in the candidate

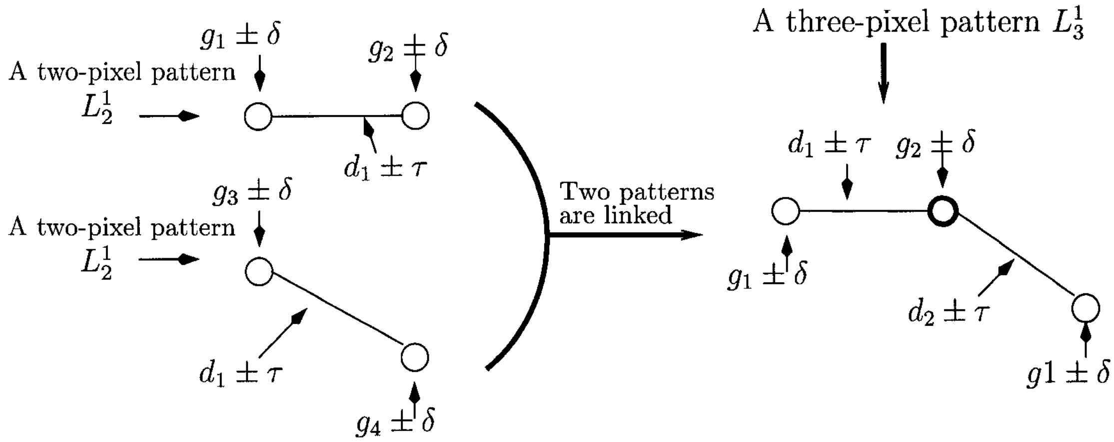


Figure 5.3: An example of the linking step

three-pixel itemset C_3 (see Table 5.5) can be represented as the following format:

$\{g_{w_x}, g_{w_y}, \theta_{i_1}, d_{j_1}, g_{w_x}, g_{w_z}, \theta_{i_2}, d_{j_2}\}, \{g_{w_x}, g_{w_y}, \theta_{i_1}, d_{j_1}, g_{w_z}, g_{w_x}, \theta_{i_2}, d_{j_2}\}$

$\{g_{w_y}, g_{w_x}, \theta_{i_1}, d_{j_1}, g_{w_x}, g_{w_z}, \theta_{i_2}, d_{j_2}\}, \{g_{w_y}, g_{w_x}, \theta_{i_1}, d_{j_1}, g_{w_z}, g_{w_x}, \theta_{i_2}, d_{j_2}\}$

$\{g_{w_y}, g_{w_x}, \theta_{i_1}, d_{j_1}, g_{w_y}, g_{w_z}, \theta_{i_2}, d_{j_2}\}, \{g_{w_y}, g_{w_x}, \theta_{i_1}, d_{j_1}, g_{w_z}, g_{w_y}, \theta_{i_2}, d_{j_2}\}$

$\{g_{w_y}, g_{w_x}, \theta_{i_1}, d_{j_1}, g_{w_y}, g_{w_z}, \theta_{i_2}, d_{j_2}\}, \{g_{w_x}, g_{w_y}, \theta_{i_1}, d_{j_1}, g_{w_z}, g_{w_y}, \theta_{i_2}, d_{j_2}\}$

Where $g_{w_x}, g_{w_y}, g_{w_z} \in \{g_{p_x}\}$

After itemset C_3 has been discovered, the sample is scanned in order to find the frequency of patterns in C_3 and only those patterns that exceed threshold t_3 are maintained in the frequent three-pixel itemset L_3 in Table 5.6. A similar procedure can be applied in order to find other multi-pixel patterns such as four-pixel patterns,

Table 5.5: Candidate three-pixel itemset C_3

$\{g_{w_1}, g_{w_2}, 0^o, d_4, g_{w_1}, g_{w_3}, 90^o, d_8\}$
$\{g_{w_1}, g_{w_2}, 0^o, d_3, g_{w_3}, g_{w_1}, 90^o, d_8\}$
$\{g_{w_2}, g_{w_1}, 0^o, d_5, g_{w_1}, g_{w_3}, 45^o, d_3\}$
$\{g_{w_1}, g_{w_2}, 0^o, d_6, g_{w_2}, g_{w_3}, 135^o, d_9\}$
$\{g_{w_1}, g_{w_2}, 45^o, d_3, g_{w_1}, g_{w_3}, 90^o, d_8\}$
....

five-pixel patterns, and so on depending on users' choices on the value of the threshold. In this thesis, three-pixel patterns are determined.

Table 5.6: Frequent three-pixel itemset L_3

....
$\{g_{w_1}, g_{w_2}, 0^o, d_4, g_{w_1}, g_{w_3}, 90^o, d_8\}$
$\{g_{w_2}, g_{w_1}, 0^o, d_5, g_{w_1}, g_{w_3}, 45^o, d_3\}$
$\{g_{w_1}, g_{w_2}, 45^o, d_3, g_{w_1}, g_{w_3}, 90^o, d_8\}$
....

5.3 Common Distinctive Patterns

The distinctive patterns discovered using this technique are extracted from one sample of the original textured image. In order to find common distinctive patterns which can represent the original textured image, more samples are required in order to find those patterns. Therefore, ten samples from the original textured image have been extracted randomly and these random samples will be scanned in the distinctive patterns found previously in order to obtain the five most common and frequent patterns. These five patterns will be served as the feature of this textured image. For any sample, if it contains three out of five common distinctive patterns, it will be classified as the texture class that contains these common distinctive patterns.

5.4 Summary of Pattern Knowledge Discovery

As introduced before, the entire procedure of the pattern knowledge discovery consists of two stages: the first step is to find distinctive patterns and the second step is to find common ones among distinctive patterns. The following six steps have shown the first stage of finding distinctive patterns for a textured image of Figure 5.4.

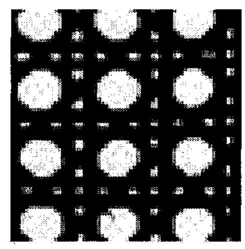


Figure 5.4: A sample of textured image with the size of 64×64

1. In the first iteration of the algorithm, each pixel of the textured image is a member of the set of candidate one-pixel pattern, C_1. L_1 contains all gray levels that exceed the predefined occurrence threshold of t_1. This step can be shown in the Figure 5.5.

2. To discover the set of distinctive patterns, L_2, the algorithm uses $L_1 \times L_1$ to generate a candidate set of two-pixel patterns, C_2.

3. Next, the original sample is scanned to determine the occurrence of those patterns in C_2. Moreover, those redundant patterns in C_2 are removed as well. Here, redundant patterns refer to patterns which are similar to each other.

4. The set of distinctive two-pixel patterns L_2 is then determined, which encompasses those candidate two-pixel patterns in C_2 having the minimum threshold t_2. Figure 5.6 shows this step.

5. The generation of the set of candidate three-pixel patterns, C_3, is similar to the technique of C_2 as $C_3 = L_2 \times L_2$.

6. The original textured image is scanned in order to determine L_3, consisting of those candidate three-pixel patterns in C_3 having the minimum threshold t_3. Figure 5.7 illustrates this step.

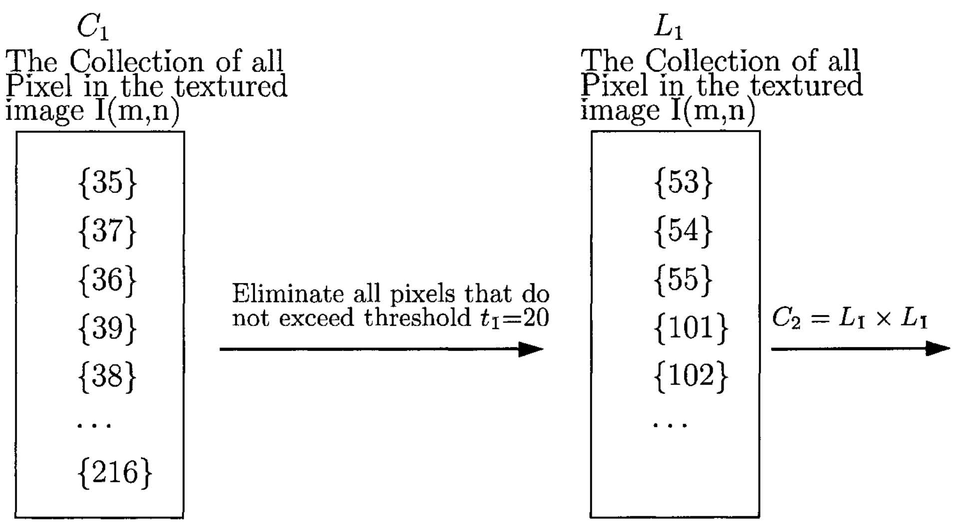


Figure 5.5: Candidate patterns set C_1 and frequent patterns set L_1

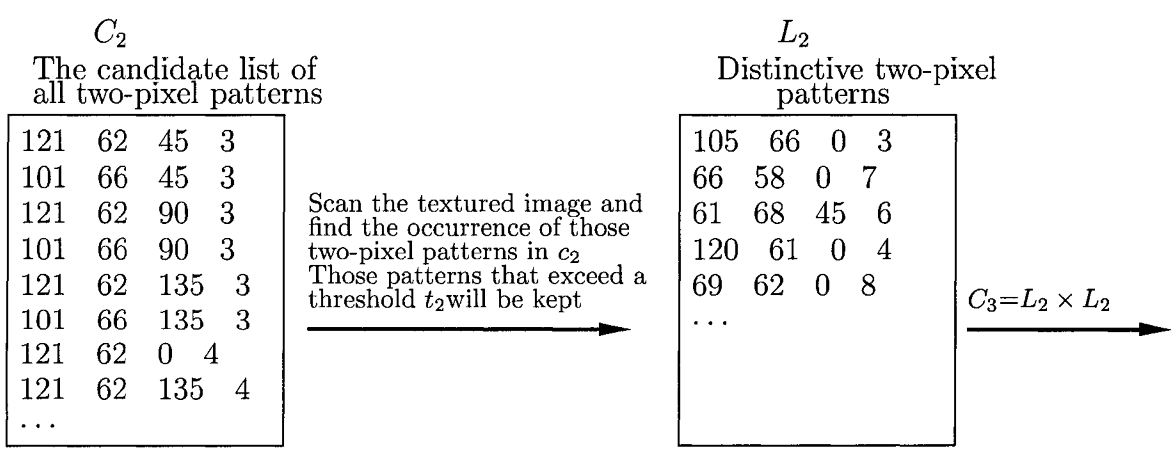


Figure 5.6: Candidate patterns set C_2 and frequent patterns set L_2

C_3
The candidate list of all three-pixel patterns

105	66	0	3	66	58	0	7
105	66	45	3	66	58	0	7
105	66	45	3	66	62	0	8
120	61	0	4	61	68	45	6
105	66	0	3	66	62	0	8
...							

Scan the textured image and find the occurrence of those three-pixel patterns in C_3 Those patterns that exceed a threshold t_3 will be kept $\longrightarrow$

L_3
Distinctive Three-pixel patterns

105	66	0	3	66	58	0	7
105	66	0	3	66	62	0	8
105	66	45	3	66	58	0	7
120	61	0	4	61	68	45	6
105	66	45	3	66	62	0	8
...							

Figure 5.7: Candidate patterns set C_3 and frequent patterns set L_3

After distinctive patterns have been discovered, the second stage of finding common frequent patterns is started. Ten samples are randomly selected in order to obtain the five most common and frequent patterns. Figure 5.8 shows one dominant feature pattern inside the textured image. Figure 5.9 shows the occurrence of this feature pattern within the ten samples of textured image.

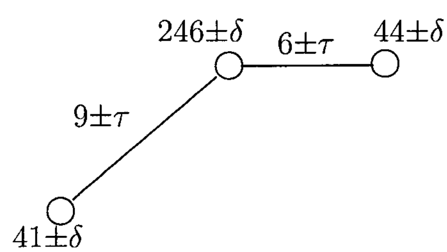


Figure 5.8: One common frequent feature pattern

Figure 5.9[1] indicates that this feature pattern is a common distinctive feature of this texture class. Figure 5.10 shows the occurrence of this feature pattern inside another 29 textured images. From the Figure 5.10, it clearly shows that D20 has 4 feature patterns and D102 has 3 feature patterns. The other 27 textured images have

[1]feature pattern can be showed distinctively on the color format

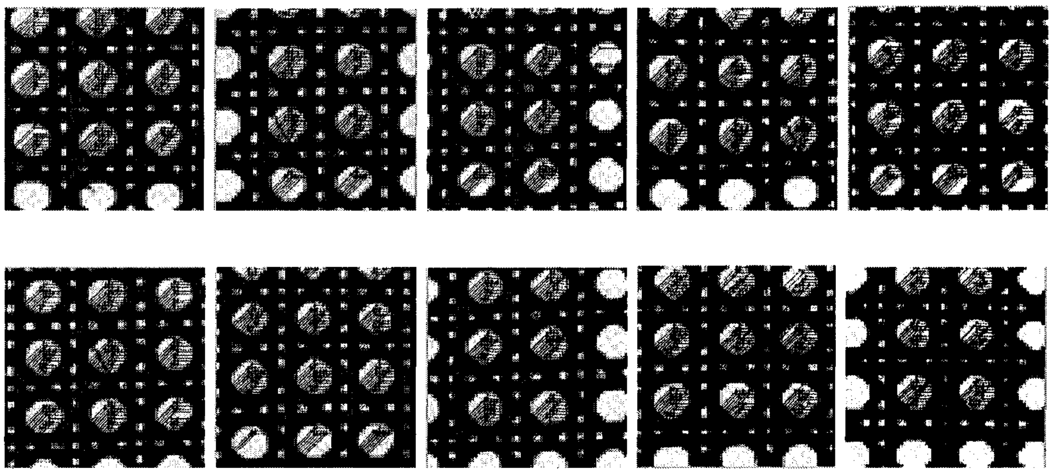


Figure 5.9: The ten samples of same textured iamge with feature patterns

no occurrence of this feature pattern. Since 3 or 4 are below the predefined threshold, they are considered as trivial occurrences. Therefore, this feature pattern can be used to classify textured images because only textures that belong to this texture class show dominant occurrences while other texture classes show trivial occurrences. In the appendix, figures from Figure B.1 to Figure B.17have shown another six examples.

5.5 Confuse Table and Other Parameters

The previous procedure of pattern knowledge discovery is applied to all known texture classes in the appendix Figure A.1. Ten training samples have been randomly selected from each texture class. After five frequent common distinctive patterns from each known texture class are discovered, training samples of each texture class are scanned to find whether they contain these common distinctive patterns. If a training sample contains three out of five frequent common distinctive patterns of a texture class, this training sample is classified to this texture class. This process relies on seven

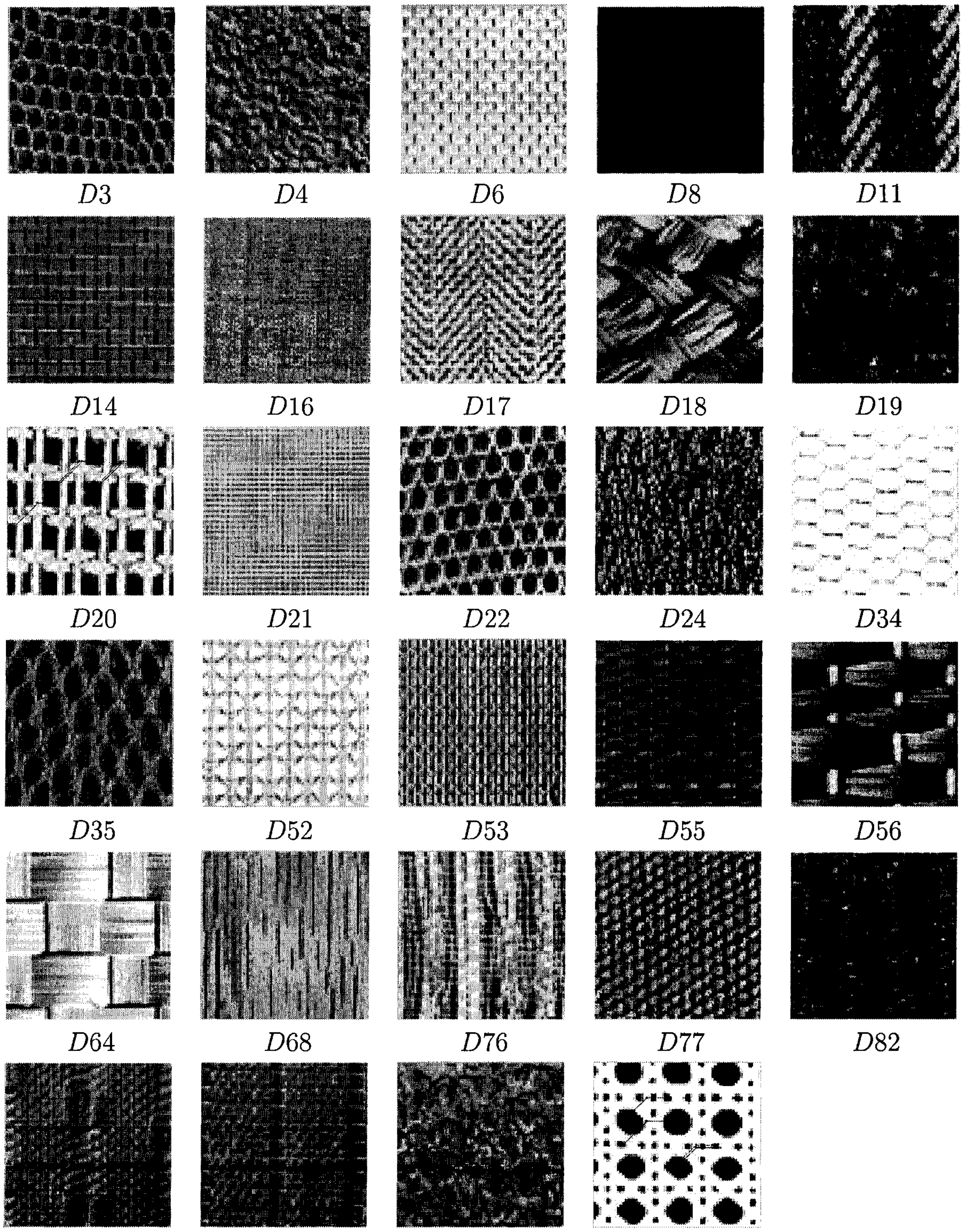


Figure 5.10: Occurrence of this feature pattern within other 29 textured images

parameters and a confuse table will be created for classification after the completion of this process.

There are three sets of parameters that have been employed in order to complete this knowledge process.

The first set consists of two parameters: τ and δ. τ stands for the distance deviation and δ stands for the gray-level deviation. The purpose of this set is to add a certain level of deviations into discovered feature patterns in order to enhance their anti-noise capabilities.[2] Normally, τ is set to 3 and δ is set to 3.

The second set consists of three parameters: t_1, t_2 and t_3. The goal of t_1 is to remove background from two-pixel candidate set C_2. For any two-pixel pattern, if the absolute value of the difference of two pixel values is less than t_1, these two pixels are then considered as the background. Figure 5.11 shows patterns that belong to the background. Suppose t_1 is 10, the absolute value of the difference between these two pixel values is 2, which is less than t_1. After two-pixel patterns from background have

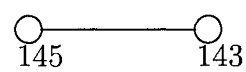


Figure 5.11: Two-pixel pattern that belongs to the background

been removed, patterns with the same direction, distance and similar pixel value will be removed in order to reduce redundancy. The parameter of t_2 will then be used to implement this goal. Figure 5.12 illustrates an example. Suppose t_2 is 10. In Figure 5.12, these two patterns are considered as similar patterns because they have the same direction and distance and the absolute value of the difference on each pair

[2]In reality, pictures can be changed easily by the environment. This capability can allow picture to resist these changes in order to keep its original state

Figure 5.12: Two similar two-pixel patterns

of pixel is less than t_2.

The parameter of t_3 is used to reduce the similarity of three-pixel patterns. Figure 5.13 illustrates an example. Suppose t_3 is 10. In Figure 5.13, these two patterns

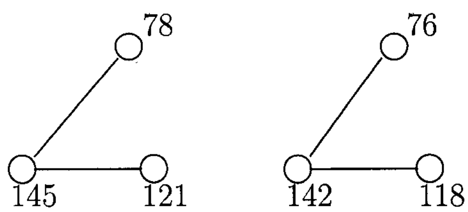


Figure 5.13: Two similar three-pixel patterns

are considered as similar patterns because they have the same direction and distance, and the absolute value of the difference on each pair of pixel is less than t_3.

The last set of parameters is made up of two parameters: n_2 and n_3. This set of parameters is used during the process of discovering complex feature patterns. The parameter of n_2 is used to select patterns in the candidate frequent set C_2 and put the chosen patterns into the frequent set L_2. The parameter of n_3 is used to select patterns in the candidate frequent set C_3 and put the chosen patterns into the frequent set L_3. Table 5.7 shows the complete list of parameters for the whole 30 texture classes.

In the Table 5.7, it shows that the value of n_2 is larger than n_3. According to the apriori algorithm, complex feature patterns tend to show fewer occurrences than the less complex feature patterns. According to the characteristics of different textures, empirical parameters are set for each of the texture classes.

Table 5.7: Complete list of parameters used in the discovery proces

	τ	δ	t_1	t_2	n_2	t_3	n_3
1	4	5	10	15	140	10	50
2	5	3	10	10	65	10	40
3	3	3	10	15	240	10	100
4	3	3	10	15	300	10	75
5	3	5	15	15	120	10	40
6	3	3	10	15	250	10	160
7	3	3	20	15	200	10	100
8	5	6	10	15	180	10	70
9	5	5	15	15	150	10	50
10	3	3	20	15	180	10	60
11	3	3	40	15	180	10	120
12	4	3	10	15	300	10	120
13	4	4	20	15	130	10	80
14	3	5	10	15	55	10	40
15	4	3	20	15	300	10	100
16	4	5	20	15	190	10	35
17	3	3	20	15	300	10	90
18	5	6	20	15	170	10	50
19	5	3	10	15	260	10	75
20	5	4	15	15	180	10	50
21	4	3	20	15	230	10	200
22	4	3	20	15	220	10	110
23	7	5	45	15	180	15	90
24	5	5	20	15	150	15	50
25	5	4	20	15	150	10	80
26	5	4	10	15	200	10	65
27	4	3	20	15	200	15	80
28	3	4	10	15	150	10	52
29	3	3	20	15	300	10	200
30	3	3	30	15	350	10	100

After the parameters have been set, all texture classes have to go through the knowledge discovery process, which eventually yields the Confuse Table. Table 5.8 shows the Confuse Table. Inside this table, some textures may contain feature patterns from the other texture class because other textures' feature patterns could be less dominant than the feature patterns inside textures. Here, less dominant feature patterns refer to the patterns with less than ten times of occuence inside the texture. Therefore, in the Table 5.8, some training samples of one texture class are assigned to other texture classes.

Entry in the table represents the number of training samples that have been assigned to texture classes. For example, texture class 3 has 10 training samples. From those 10 training samples, 10 out of 10 are assigned to texture class 3 according to the feature patterns of texture class 3 , 10 out of 10 is assigned to texture class 8 according to the feature patterns of texture class 8, 1 out of 10 is assigned to texture class 12 according to the feature patterns of texture class 12, 6 out of 10 are assigned to texture class 23 according to the feature patterns of texture class 23. For 10 training samples of each texture class, if less than three training samples contain feature patterns of a specific texture class, these training samples are believed to not belong to this specific texture class. The modified confuse table is shown in the Table 5.9 which will be used for classification. In Table 5.9, if the entry in the table contains 1, it means training samples of the texture classes on the row contain the feature patterns of the texture class on the column.

Table 5.8: Confuse Table for textured image samples

Texture Classes

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
1	10												6	10					10											
2		10			2									10					10											
3			10					10				1											6							
4	4			10									10																	
5		10			10	10													7								7	10		
6		10			9	10	1							9					10								9	10		
7		4			1	5	10		1									1		1		10								
8								10										4					1							
9		3							10										6			1								
10		10			10	10				9									10						1	10	10	10		
11											10																			
12								10	10			10						1				10	10							
13													10						3											
14														10					10											
15															10		4													
16													2	4		10			6			2								
17								10									10						10							
18								1										10	2											
19		9			3									6					10							5		9		
20		1							2										10	10						9	3	1		
21								10	10			7									10	1	10							
22					4				9									8		2		10	6	6						
23								6										3					10							
24		8																6	9					10						
25		4											5	10					10						10	10	3	9		
26		10			10	10								2					10						1	10	10	10		
27		10			10	10								9					10							10	10	10		
28		10			3	10								6					10							1	3	10		
29											9																		9	9
30																														10

Table 5.9: Modified Confuse Table

Texture Classes

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
1	1												1	1					1											
2		1												1					1											
3			1					1															1							
4	1			1									1																	
5		1			1	1													1								1	1		
6		1			1	1								1					1								1	1		
7		1				1	1															1								
8								1										1												
9		1							1										1											
10		1			1	1				1									1							1	1	1		
11											1																			
12								1	1			1										1	1							
13													1						1											
14														1					1											
15															1		1													
16														1		1			1											
17								1									1						1							
18																		1												
19		1			1									1					1							1		1		
20																			1	1						1	1			
21								1	1			1									1		1							
22					1				1									1				1	1	1						
23								1										1					1							
24		1																1	1					1						
25		1											1	1					1						1	1	1	1		
26		1			1	1													1							1	1	1		
27		1			1	1								1					1							1	1	1		
28		1			1	1								1					1								1	1		
29											1																		1	1
30																														1

5.6 Classification

The procedure of classification is described as follows: An unknown test sample u is scanned in order to find the existence of common distinctive patterns for known texture classes. If the test sample u contains enough amounts of distinctive feature patterns from a known texture class, it may belong to this texture class. The amount of distinctive feature patterns has been defined as a threshold. After scanning the test sample u for feature patterns of all known texture classes, the test sample u is assigned to a texture class if the test sample u contains the feature patterns from this texture class only. If the test sample u contains the feature patterns from more than one texture class, the modified confuse table is applied to classify. For example, if the test sample u contains the feature patterns from texture classes 2 and 9, the test sample u could belong to texture class 2 or texture class 9. The modified confuse table is applied to check the properties of texture classes 2 and 9. It shows that texture class 9 contains the feature patterns from the texture class 9 while the texture class 2 does not contain these feature patterns. Therefore, the test sample u is scanned to find the occurrence of the feature patterns from texture class 9. If the sample u contains these feature patterns, it is then assigned to texture class 9; otherwise, it is assigned to texture class 2. Table 5.10 shows the excerpted part from the Modified Confuse Table 5.9.

Table 5.10: Excerpted part from the Modified Confuse Table

Assigned Texture Classes

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
2	...	1	...	...	...	...	...	...	...	...	...	...	...	1	...	...	...	...	1	...	...	...	...	...	...	...	...	...	...	...
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
9	...	1	...	...	...	...	...	...	1	...	...	...	...	...	...	...	...	...	1	...	...	...	...	...	...	...	...	...	...	...
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...

5.7 Results

In our system, 30 textures (in the appendix Figure A.1) from the benchmark database (Brodatz album [4]) have been selected and 10 random samples with a size of 64x64 have been tested for each texture. Table 5.11 shows the final result for all samples after classification. From this table, we observe that most regular textures have better accuracy than non-regular textures. This is normal because regular textures tend to have homogeneous properties more than non-regular textures. Therefore, random samples extracted from regular textures should have an increased likelihood to find common distinctive patterns than others.

From the table 5.11, 19 out of 30 textures have 9 or 10 out of 10 samples classified correctly and 5 textures have 7 or 8 out of 10 samples classified correctly and the remaining 6 textures have 5 or 6 out of 10 samples classified correctly. Reasons for the misclassified testing samples are: (1) they are not textures with enough constant patterns. As distinctive three-pixel patterns are discovered from 10 training samples, those testing samples would become significantly different from training samples if the texture does not contain constant patterns; and (2) the degree of similarity between some textures is high. For instance, D83 and D85 look very close so that it is hard to classify them correctly. D3 and D35 cannot be differentiated greatly because they are both visually similar.

It should be noticed that the number of textures employed in the testing is bigger than that of many experiments reported in this area. The large number of textures in the test surely increases the difficulty of the recognition task.

5.8 Comparison Experiments

In order to evaluate the performance of the proposed system, a strategy of making comparison experiments have been conducted. According to the strategy, the proposed system will be compared with another four popular techniques which are Co-occurrence Matrix, 2-D Artificial Autoregressive, Edge Frequency and Law's Filter method. The testing database is not restricted to Broadz Album because the VisTex database and MeasTex database are also chosen for evaluation in order to increase objectivity. VisTex and MeasTex are chosen because they are benchmark image databases in the area of texture analysis. Many researchers have used these databases to investigate their approaches.

5.8.1 Comparison with Co-occurrence Technique

The features of the co-occurrence matrix adopted for the comparison experiments are correlation, energy and homogeneity. Distances are set from 2 to 4 and features from four directions are averaged which give nine features for each texture samples in total. The classification algorithm is the nearest neighbor algorithm. Ten training samples and ten testing samples from each texture class are selected randomly. Table 5.12 shows the classification result from the co-occurrence method. The average classifica-

Table 5.11: Final results table

Assigned Texture Classes

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
1	7												2						1											
2		6												3					2											
3			10																											
4	1			9																										
5					10																									
6						9	1																							
7					5		5																							
8								7																						
9		1							6										2			1								
10										9																1				
11											7																			
12												9									1									
13													10																	
14														10																
15															10															
16	4												1			5														
17																	10													
18														1				9												
19		1																	9											
20									2											8										
21																					10									
22																						10								
23																							10							
24																								10						
25																									10					
26																										5	5			
27																										1	9			
28					1	2																				1		6		
29											2																		8	
30																														10

1. Reptile Skin (d3)
2. Pressed cork(d4)
3. Woven aluminum wire (d6)
4. Abstract illusion of woven wire(d8)
5. Homespun woolen cloth (d11)
6. Woven aluminum wire (d14)
7. Herringbone weave (d16)
8. Herringbone weave (d17)
9. Raffia weave (d18)
10. Woolen cloth (d19)
11. French canvas (d20)
12. French canvas (d21)
13. Reptile skin (d22)
14. Pressed calf leather(d24)
15. Netting (d34)
16. Lizard skin (d35)
17. Oriental straw cloth (d52)
18. Oriental straw cloth (d53)
19. Straw matting flat lighting (d55)
20. Straw matting (d56)
21. HandwovenOriental rattan (d64)
22. Wood grain (d68)
23. Oriental grass fiber cloth (d76)
24. Cotton canvas (d77)
25. Oriental straw cloth (d82)
26. Woven matting (d83)
27. Straw matting with bamboo(d85)
28. Pigskin (d92)
29. Cane (d101)
30. Cane (d102)

Table 5.12: Classification result from Co-occurrence method

Assigned Texture Classes

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
1	7	2																	1											
2		6												4																
3			10																											
4				10																										
5					9																		1							
6						10																								
7							10																							
8								10																						
9					1				8											1										
10					5					3													1					1		
11											10																			
12												10																		
13													10																	
14		1												9																
15																	10													
16																10														
17		2															8													
18																		10												
19		5												1					4											
22									1											9										
21																					10									
22																						10								
23																							10							
24																								10						
25																									10					
26					1																		5			4				
27																									1		9			
28																												10		
29																													10	
30																														10

tion rate from the co-occurrence matrix technique is 85% while the proposed system is 84%, which is slightly lower. Table 5.13 gives the detailed information about the classification rate on each texture class. For each texture class, the classification rate of proposed system and co-occurrence technique has been displayed. From this table, it shows that most texture classes have almost equal rate except texture no.15. For the texture no.15, which is d34 in the Brodatz album, the proposed system has 100% classification rate while the co-occurrence technique has a nil classification rate. This clearly shows that the proposed system is superior to the co-occurrence technique in dealing with textures with regular patterns.

Table 5.13: The comparison between proposed system and co-occurrence method

	P	C		P	C		P	C		P	C		P	C		P	C
1	0.7	0.7	6	0.9	1	11	0.7	1	16	0.5	1	21	1	1	26	0.5	0.4
2	0.5	0.6	7	0.5	1	12	0.9	1	17	1	0.8	22	1	1	27	0.9	0.9
3	1	1	8	0.7	1	13	1	1	18	0.9	1	23	1	1	28	0.6	1
4	0.9	1	9	0.6	0.8	14	1	1	19	0.9	0.4	24	1	1	29	0.8	1
5	1	0.9	10	0.9	0.3	15	1	0	20	0.8	0.9	25	1	1	30	1	1

5.8.2 Comparison with 2-D AR Model

The 2-D AR model for texture classification [43] was chosen for 12 textures (Appendix Figure A.2). The same texture classes are selected and the same numbers of testing samples are provided. The results in the table 5.14 showed that the overall performance is slightly different. However, for d21 texture, our proposed system can get 9 out of 10 testing samples classified correctly while the method based on the neural network can only classify 5 out of 10 testing samples correctly.

Table 5.14: The comparison between 2-D AR approach and proposed system

	1	2	3	4	5	6	7	8	9	10	11	12
1	7(9)											
2		9(10)										
3			9(5)									
4				10(10)								
5					10(10)							
6						9(10)						
7							9(10)					
8								8(10)				
9									10(10)			
10										10(10)		
11											10(10)	
12												10(10)

1. Herringbone weave (d17) 2. Woolen cloth (d19)
3. French canvas (d21) 4. Netting (d34)
5. Oriental straw cloth (d52) 6. Oriental straw cloth (d53)
7.Straw matting flat lighting (d55) 8. Straw matting (d56)
9.HandwovenOriental rattan (d64) 10. Wood grain (d68)
11. Oriental grass fiber cloth (d76) 12.Cotton canvas (d77)

Note: The figure inside the parenthesis of the table is the result of using 2-D AR technique.

5.8.3 Comparison with Law's Method

For Law's method, a total of 25 masks are convolved with the image to detect different features such as linear elements and ripples. These masks have been proposed by Law's [19]. In [29], they compute five amplitude features for each convolution, namely mean, standard deviation, skewness, kurtosis, and energy measurement. The classification algorithm is linear classifier for it gives the best performance. The benchmark database is MeasTex, which is a popular public texture database and can be found in the Appendix Figure C.1. The classification result of the Law's method in the MeasTex database is 82.8%. Table 5.15 shows the parameters when textures in the MeasTex database have been applied into our proposed system. Table 5.16 shows the classification result.

Table 5.15: Parameters used for MeasTex database

	τ	δ	t_1	t_2	n_2	t_3	n_3
Asphalt	3	3	51	20	200	15	150
Concrete	2	3	10	15	250	10	160
Grass	3	3	10	15	200	10	100
Rock	4	3	10	15	130	10	60

Table 5.16: Classification result of MeasTex

	Assigned Texture Class			
	Asphalt	Concrete	Grass	Rock
Asphalt	10	0	0	0
Concrete	1	9	0	0
Grass	0	0	10	0
Rock	0	0	0	8

From Table 5.16 ,the total classification rate is 90%, which is higher than the Law's method. The reason is because these textures all contain strong regular patterns.

5.8.4 Comparison with Edge Frequency Method

According to [29], they compute the gradient difference between a pixel f(x,y) and its neighbors at a distance d. For a given value of distance, the gradient differences can be summed up over the whole image. For different values of d, different feature measurements for the same image have been obtained. The classification algorithm is kNN and k is set to 3 empirically. The benchmark database is the public VisTex database which can be found in the appendix Figure C.2. The classification rate of Edge Frequency method is 66.8% while the proposed system gives the classification rate of 62.8%. Table 5.17 shows the parameters when VisTex has been applied to the proposed system and Table 5.18 shows the detailed classification result of VisTex.

From Table 5.18 , it indicates that the proposed system does not perform well on

Table 5.17: Parameters for VisTex

	τ	δ	t_1	t_2	n_2	t_3	n_3
Bark	3	3	10	15	100	10	70
Fabric	2	3	10	15	230	10	100
Food	4	3	10	15	80	10	35
Metal	6	3	10	15	50	10	35
Sand	3	3	20	20	200	20	65
Tile	3	3	10	15	210	10	100
Water	4	3	10	15	100	10	50

Table 5.18: Classification result for VisTex

	Bark	Fabric	Food	Metal	Sand	Tile	Water
Bark	3	1	0	0	3	0	3
Fabric	0	9	0	0	0	1	0
Food	0	2	7	0	0	0	0
Metal	0	1	0	7	0	0	2
Sand	0	0	0	0	8	0	2
Tile	0	5	1	0	0	4	0
Water	3	1	0	0	0	0	5

Bark, Tile and Water. Detailed investigation reveals that these three textures contain little or no regular patterns while other texture classes with high classification rates contain much more regular patterns.[3] This again confirms that our proposal system excels in dealing with textures with regular patterns.

5.8.5 Summary of Comparison Experiments

In order to evaluate the proposed system, four popular approaches in the texture analysis have been employed and three public texture databases have been involved. Results showed that the proposed system has almost equal classification rate with co-occurrence matrix technique and 2-D AR Model method. However, the proposed

[3]The texture of tile is considered as irregular because its random images with the window size contain inconsistent regular patterns

system distinguished itself from performing well on textures with regular patterns. For MeasTex database, the proposed system has been compared with the Law's Method and showed a higher classification rate. When the VisTex database has been employed, both proposed system and edge frequency failed to yield satisfying results. Detailed investigation has shown that our proposed system is dealing well with the regular textures while it gives poor results to the textures with irregular patterns. This phenomenon is under the normal range because if the texture does not contain enough regular patterns, the extracted feature patterns cannot represent the texture class well and hence lead to the poor performance.

Chapter 6

Summary and Future Research

6.1 Summary of Contributions

In this thesis, an approach of texture classification which uses the knowledge discovery process to find distinctive patterns has been studied. An original system has been developed to perform texture feature extraction for texture classification. This system consists of two steps for the texture feature extraction: the first step is to find distinctive patterns within one training sample of the texture class and the second step is to find common distinctive patterns among the rest of the training samples of this texture class.

In the first step of finding distinctive texture patterns, the Apriori algorithm guides the procedure in order to conduct the level-wise search. Complex patterns are discovered through less complex patterns and predefined thresholds have been used in order to reduce the searching space.

In the second step, distinctive texture patterns from the first step have to go through the rest of the training samples in order to find out common distinctive patterns. The discovered common distinctive patterns have the following characteristics: (1) they are anti noise. Because textures in reality are subject to all kinds of noises, grey level deviations and distance deviations are added in order to find strong texture patterns that can handle noise issues; and (2) common distinctive patterns can be used repeatedly in order to classify unknown test samples.

In the learning procedure, all training samples have been scanned according to the common distinctive texture patterns of each texture class. If one sample contains three out of five common distinctive texture patterns, it is then classified into the corresponding texture class. This procedure also yields a modified confuse table that can be used to help determine samples containing more than one texture class's feature patterns.

A series of experiments were conducted to test the performance of the proposed texture classification system on natural textured images. A classification problem of thirty natural textures provided by the Brodatz album was considered to demonstrate the ability of the proposed system in classification and highly satisfactory results were achieved for most of the textures involved in the test phase.

The proposed system used a systematic way to find the distinctive patterns of textured images and the discovered texture patterns is scalable, which means higher complex patterns can be discovered from existing patterns and it depends on the users to determine the level of complexity for texture patterns.

In order to evaluate the performance of the proposed system, comparison experiments have been organized systematically. Four popular approaches with two other public texture databases have been employed. Results have proven the superior capability of the proposed system in dealing with textures with regular patterns.

6.2 Future Work

For the future work, the emphasis should be put on the classification of broad categories of textures. For example, for textures of lizard skin and snake skin, the feature patterns of these two categories can be combined together in order to find the feature patterns of the reptile skin which include skins of lizard and snake. The potential tricky part would be the level of similarity between the categories. If the level of similarity is high , the classification result may not become encouraging.

As images in RGB format are made up of three matrices which correspond to one of the colors red, green or blue, for each matrix, the proposed system might be applied in order to find the feature patterns. Together, feature patterns for this RGB image can be determined.

Bibliography

[1] *VisTex database.* http://vismod.media.mit.edu/vismod/imagery/VisionTexture, Accessed on August 22,2006.

[2] *The free dictionary.* http://www.thefreedictionary.com/texture, Accessed on November 2,2006.

[3] *Meastex databse.* http://www.texturesynthesis.com/meastex/meastex.html, Accessed on September 24,2006.

[4] P. Brodatz. *Textures: A Photographic Album for Artists & Designers.* Dover Publications, second edition, 1966.

[5] C.E.Brodley and P.E.Utgoff. Multivariate decision trees. *Machine Learning*, 19(1):45–77, 1995.

[6] R. R. Coifman and M. V. Wickerhauser. Wavelets and adapted waveform analysis. In *Studies in Advanced Mathematics*, pages 399–423, 1992.

[7] D. D.Michie and C.C.Taylor. *Machine Learning, Neural and Statistical Classification.* Ellis Horwood, 1994.

[8] R. Duda and P. Hart. *Pattern Classification and Scene Analysis.* John Wiley&Sons, 1973.

[9] K. Fu. *Syntactic Pattern Recognition and Applications.* Prentice Hall, 1982.

[10] M. M. Galloway. Texture analysis using grey level run lengths. *Computer Graphics and Image Processing*, 4:172–179, 1975.

[11] R. C. Gonzalez and R. E. Woods. *Digital Image Processing.* Prentice Hall, second edition, 2002.

[12] R. C. Gonzalez and R. E. Woods. *Digital Image Processing.* Pearson Education, Inc, second edition, 2004.

[13] R. Haralick and L. Watson. A facet model for image data. *Computer Vision Graphics Image Process*, 15:113–129, 1981.

[14] W. Inmon. *Building the Data Warehouss.* John Wiley&Sons, 1996.

[15] M. James. *Classification Algorithms.* John Wiley&Sons, 1985.

[16] H. Kaizer. A quantification of textures on aerial photographs. In *Technical Note 121, AD 69484*, 1955.

[17] D. Keim. *Visual techniques for exploring databases.* In Tutuorial Notes, 3rd Int. Conf. Knowledge Discovery and Data Mining, 1997.

[18] K.Fukunaga and D.Hummels. Bayes error estimation using parzen and k-nn procedure. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 9(5):634–643, 1987.

[19] K. Laws. *Textured image segmentation.* PhD Thesis, University of Southern California, 1980.

[20] P. L.v.Gool and A. Oosterlink. Texture analysis anno 1983. *Computer Vision, Graphics, and Image Processing*, 29:336–357, 1985.

[21] A. C. B. M. Clark and W. S. Geisler. Texture segmentation using gabor modulation/ demodulation. *Pattern Recognition Letters*, 6:261–267, 1987.

[22] D. Pyle. *Data Preparation for Data Mining.* Morgan Kaufmann, 1999.

[23] K. S. R. Haralick and I. Dinstein. Textural features for image classification. *IEEE Transactions on Systems, Man, and Cybernetics*, 3(6):610–621, 1973.

[24] A. S. Rakesh Agrawal, Tomasz Imielinski. Mining association rules between sets of items in large databases. In *The ACM SIGMOD International Conference on Management of Data*, pages 207–216, 1993.

[25] R.Chellappa and R.L.Kashyap. Texture synthesis using 2-d non-casual autoregressive models. *IEEE Trans. Acoust. Speech Signal Process*, ASSP-33:959–963, 1985.

[26] B. V. R. C. R. R.L. Kennedy, Y. Lee and R. Lippman. *Solving Data Mining Problems through Pattern Recognition.* Prentice Hall, 1998.

[27] R.L.Dobrusin. Markov processes with a large number of locally interacting components: Existence of a limit process and its ergodicity. *Probl. Peredachi Inform.*, 7(2):70–87, 1971.

[28] R.M.Haralick. Statistical and structural approaches to texture. *Proceedings of the IEEE*, 67(5):786–804, 1979.

[29] S. Singh and M. Sharma. Texture analysis experiments with meastex and vistex benchamarks. In *the Second International Conference on Advances in Pattern Recognition*, pages 417–424, 2001.

[30] P. Souza. Texture recognition via autoregression. *Pattern Recognition*, 15(6):471–475, 1982.

[31] W.-Y. L. T.-S. Lim and Y.-S. Shih. A comparison of prediction accuracy, complexity, and training time of thirty-three old and new classification algorithms. *Machine Learning*, 40(3):203–228, 1995.

[32] T.Cover and P.Hart. Nearest neighbor pattern classification. *IEEE Trans. Information Theory*, 13:21–27, 1967.

[33] M. G. Thomson and D. H. Foster. Role of second- and third-order statistics in the discriminability of natural images. *Journal of the Optical Society of America A: Optics, Image Science, and Vision*, 14(9):2081–2090, 1997.

[34] F. Tomita and S. Tsuji. *Computer analysis of visual textures.* Kluwer Academic Publishers, 1990.

[35] T. Toyoda and O. Hasegawa. Texture classification using extended higher order local autocorrelation features in texturen. In *The 4th International Workshop on Texture Analysis and Synthesis*, pages 131–136, 2005.

[36] Tuceryan and Jain. Texture analysis. In *Handbook of Pattern Recognition and Computer Vision*, pages 235–276, 1993.

[37] S. Weiss and C.A.Kulikowski. *Computer systems that learn: classification and prediction methods from statistics, neural nets, machine learning, and expert systems.* Morgan Kaufmann Publishers Inc., 1991.

[38] S. Weiss and N. Indurkhya. *Predictive data Mining.* Morgan Kaufmann, 1998.

[39] C. Westphal and T. Blaxton. *Data Mining Solutions: Methods and Tools for Solving Real-World Problems.* John Wiley&Sons, 1998.

[40] D. C. R. R. A. Wezska, J.S. A comparative study of texture measures for terrain classification. *IEEE Transactions on Systems, Man, and Cybernetics*, 6(4):269–285, 1976.

[41] R. D. Widrow, B. and M. Lehr. Neural networks: Applications in industry, business and science. *Communications of the ACM*, 37(3):93–105, 1994.

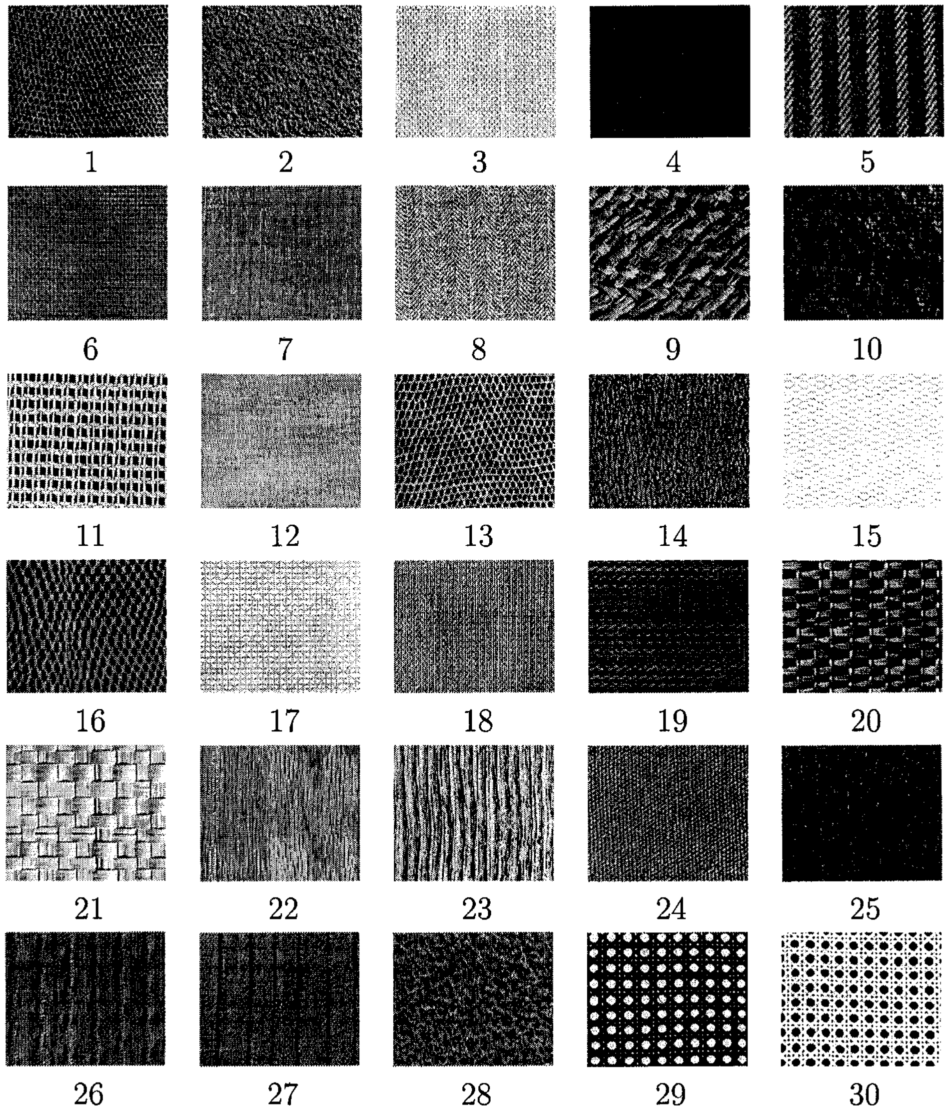
[42] J. W. Woods. Two-dimensional discrete markovian fields. *IEEE Transactions on Information Theory*, 18(2):232–240, 1972.

[43] X.He. *M.Sc. Thesis at Department of Computer Science.* Memorial University of Newfoundland, 1992.

[44] S. W. Zucker. Region growing: Childhood and adolescence. *Computer Graphics and Image Processing*, 5:382–399, 1976.

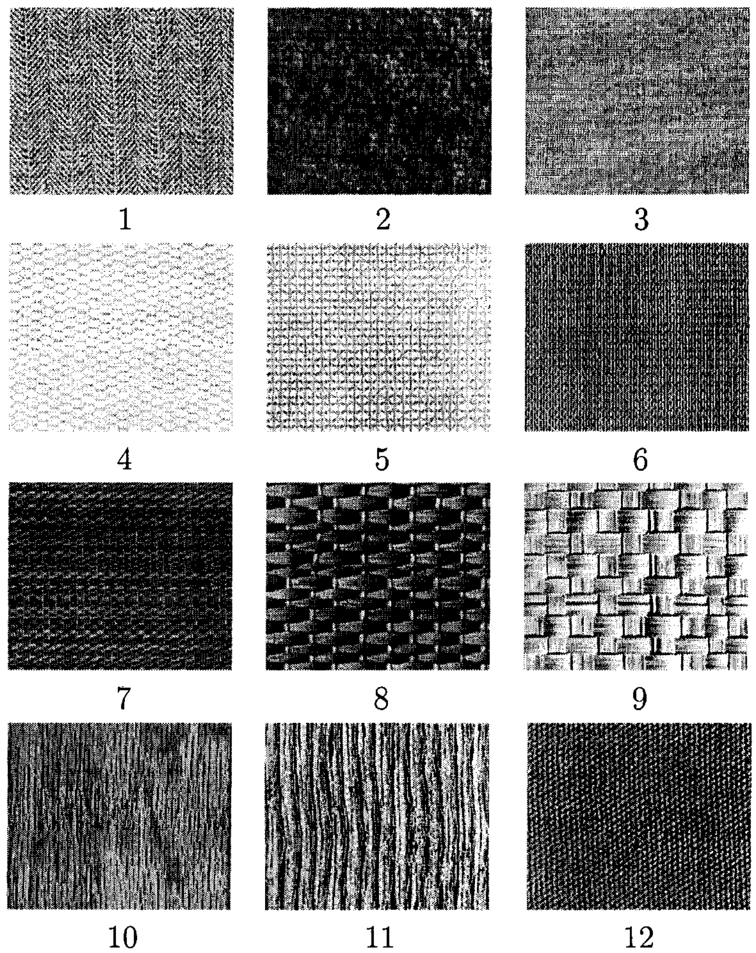
Appendix A

Textures for the experiments



1. Reptile Skin (d3)
2. Pressed cork(d4)
3. Woven aluminum wire (d6)
4. Abstract illusion of woven wire(d8)
5. Homespun woolen cloth (d11)
6. Woven aluminum wire (d14)
7. Herringbone weave (d16)
8. Herringbone weave (d17)
9. Raffia weave (d18)
10. Woolen cloth (d19)
11. French canvas (d20)
12. French canvas (d21)
13. Reptile skin (d22)
14. Pressed calf leather(d24)
15. Netting (d34)
16. Lizard skin (d35)
17. Oriental straw cloth (d52)
18. Oriental straw cloth (d53)
19. Straw matting flat lighting (d55)
20. Straw matting (d56)
21. HandwovenOriental rattan (d64)
22. Wood grain (d68)
23. Oriental grass fiber cloth (d76)
24. Cotton canvas (d77)
25. Oriental straw cloth (d82)
26. Woven matting (d83)
27. Straw matting with bamboo(d85)
28. Pigskin (d92)
29. Cane (d101)
30. Cane (d102)

Figure A.1: Texture from the Brodatz album for classification experiments.



1. Herringbone weave (d17)
2. Woolen cloth (d19)
3. French canvas (d21)
4. Netting (d34)
5. Oriental straw cloth (d52)
6. Oriental straw cloth (d53)
7. Straw matting flat lighting (d55)
8. Straw matting (d56)
9. HandwovenOriental rattan (d64)
10. Wood grain (d68)
11. Oriental grass fiber cloth (d76)
12. Cotton canvas (d77)

Figure A.2: Textures for the experiments of comparison

Appendix B

Results of some experiments

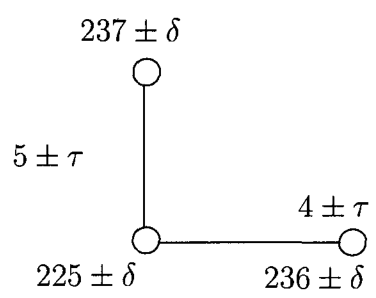


Figure B.1: One common frequent feature pattern of the texture class d6

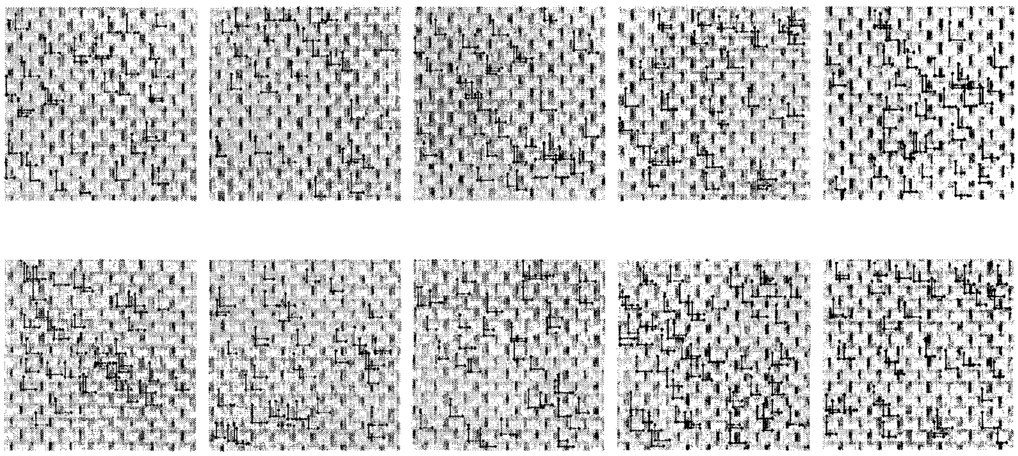


Figure B.2: Ten random samples of same textured image d6 with feature patterns

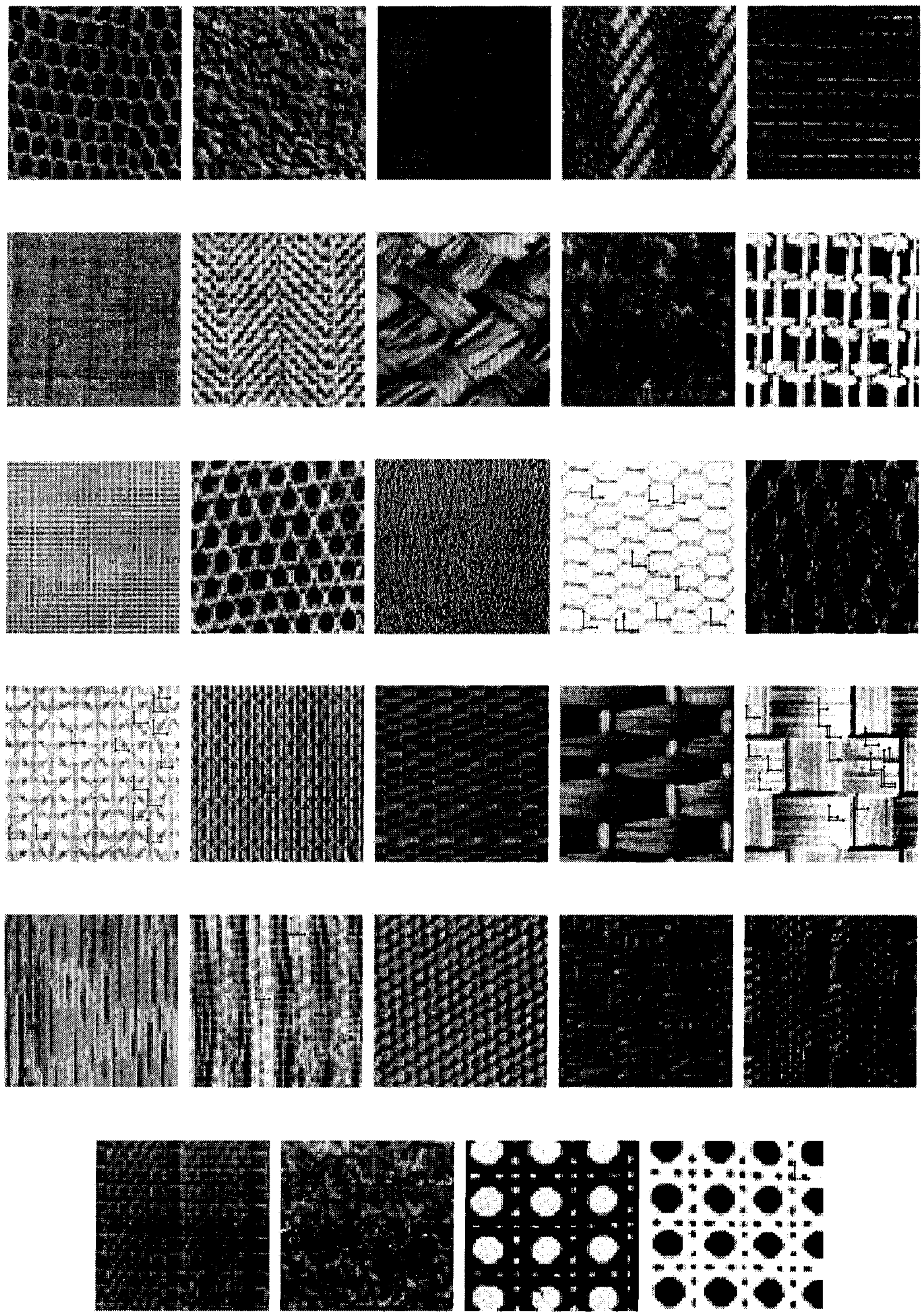


Figure B.3: Occurrence of d6 feature pattern within other 29 textured images

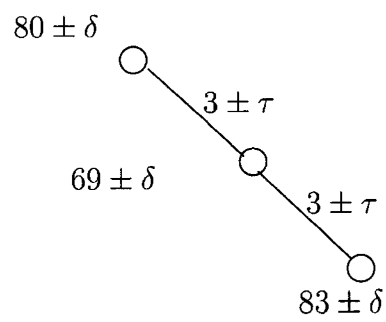


Figure B.4: One common frequent feature pattern of the texture class d8

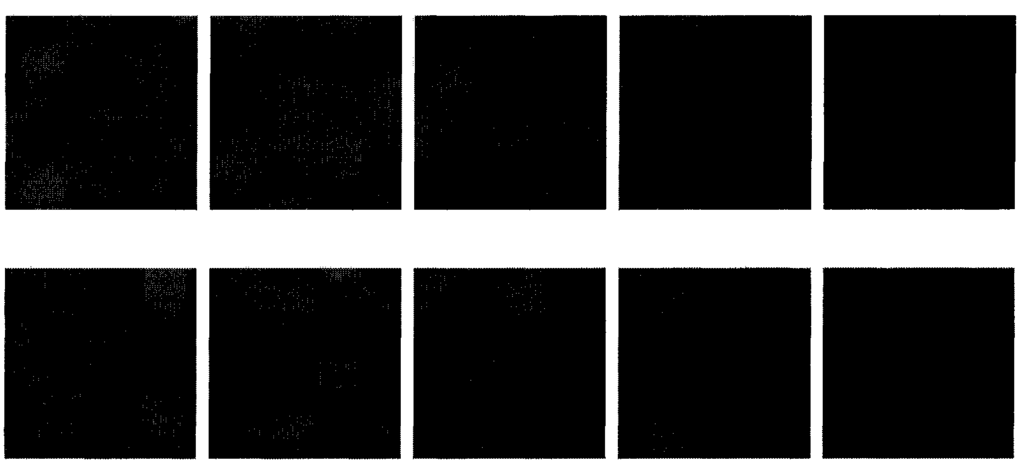


Figure B.5: Ten random samples of same textured image d8 with feature patterns

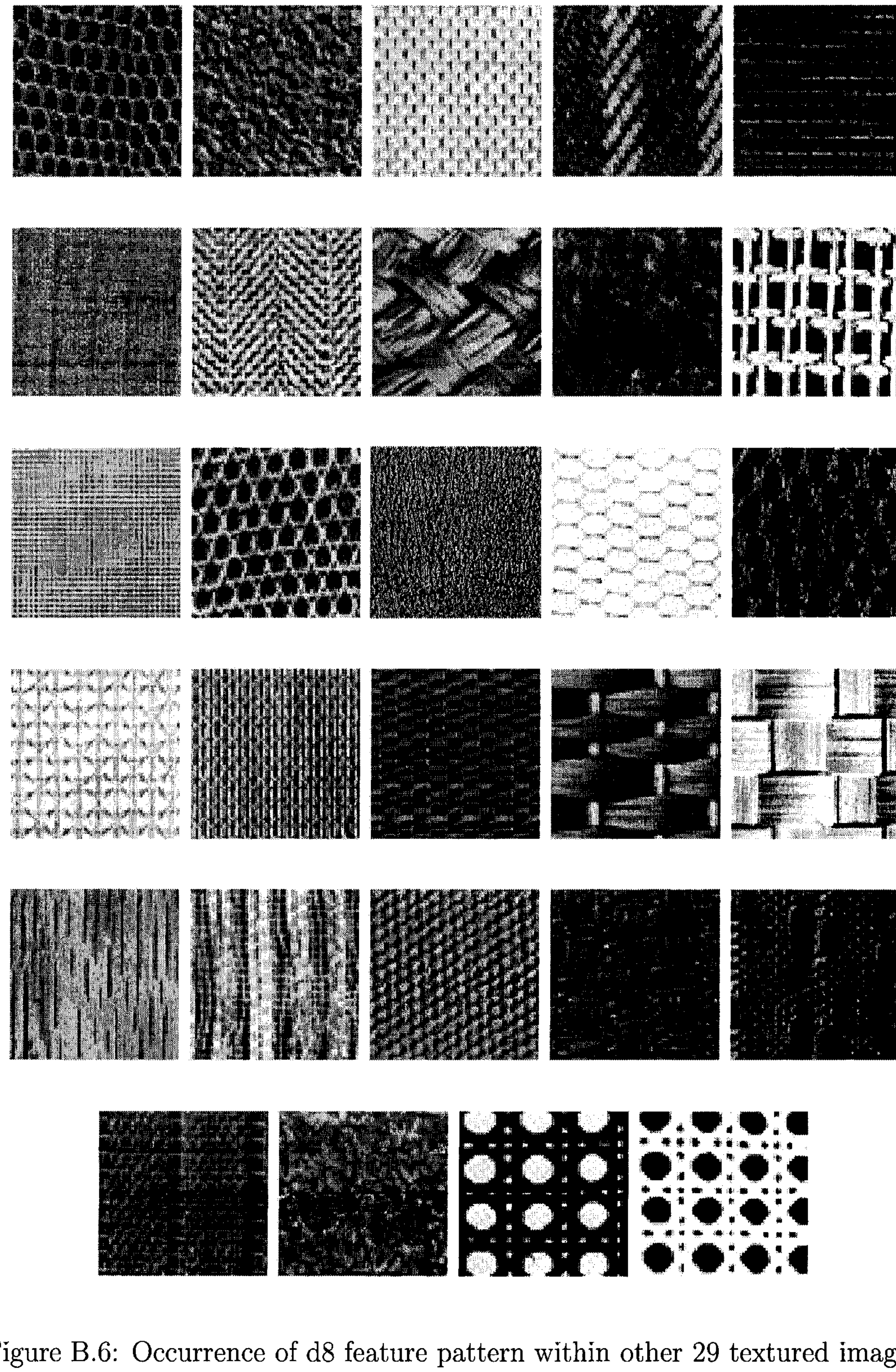


Figure B.6: Occurrence of d8 feature pattern within other 29 textured images

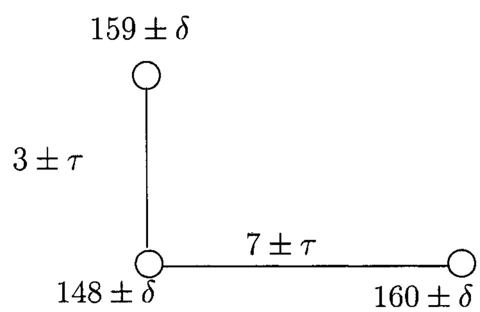


Figure B.7: One common frequent feature pattern of the texture class d16

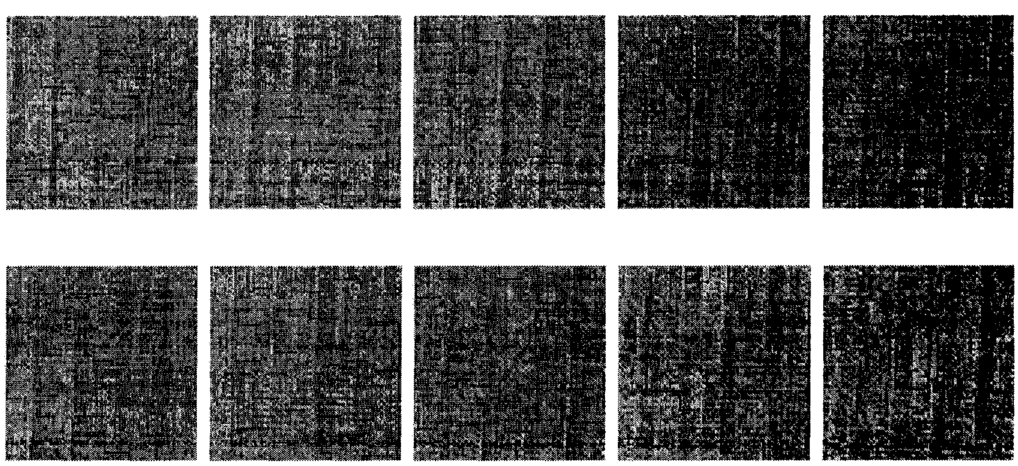


Figure B.8: Ten random samples of same textured image d16 with feature patterns

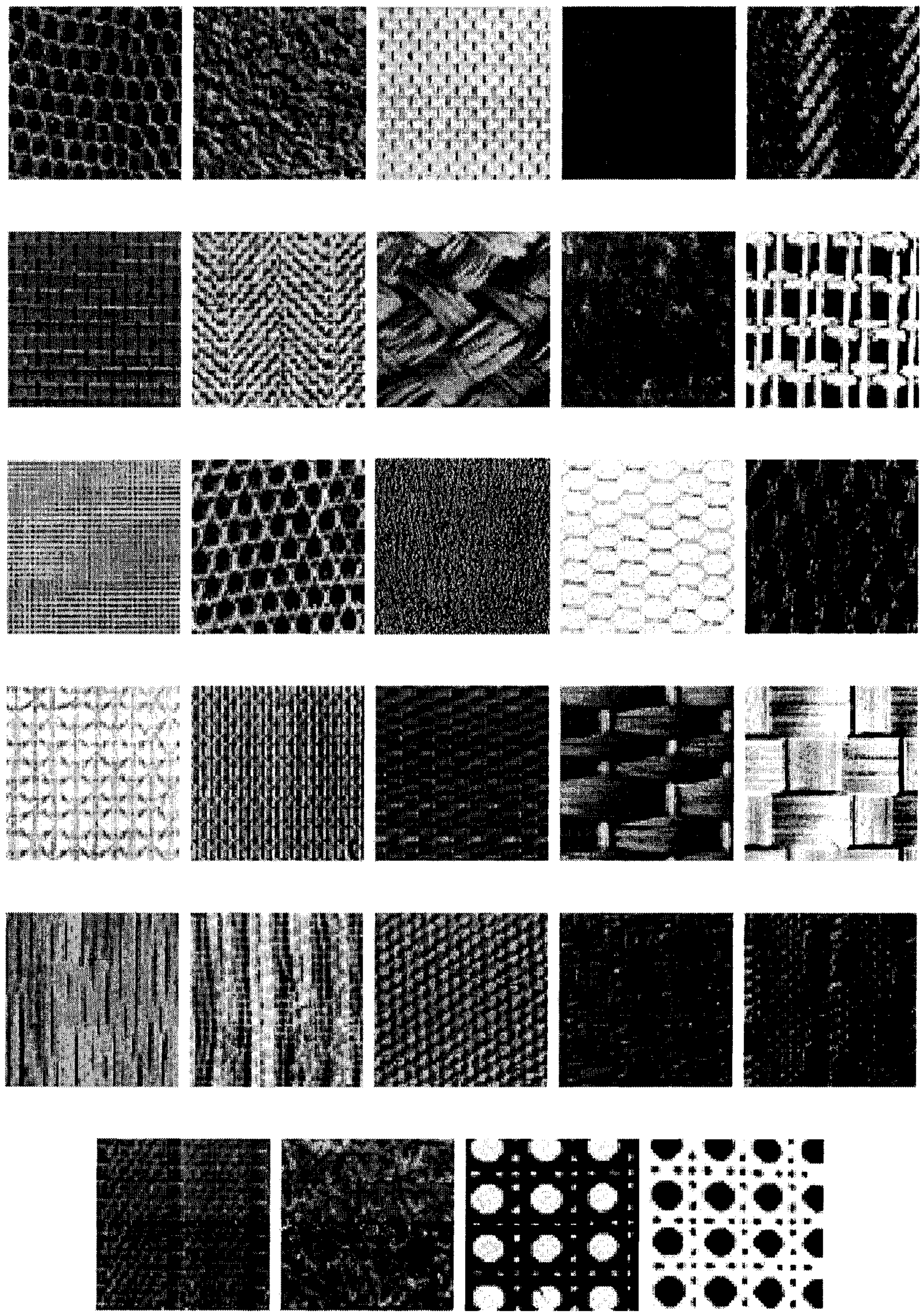


Figure B.9: Occurrence of d16 feature pattern within other 29 textured images

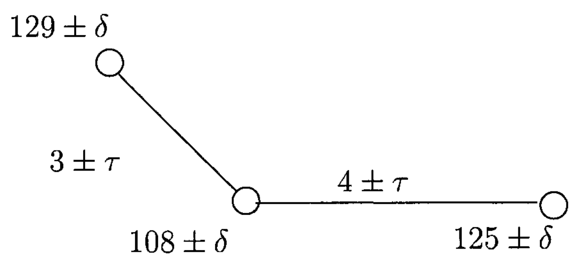


Figure B.10: One common frequent feature pattern of the texture class d19

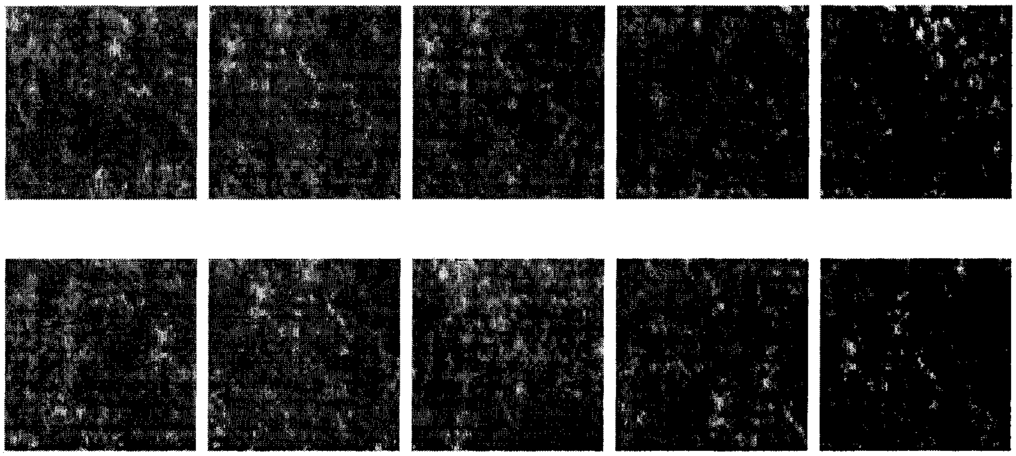


Figure B.11: Ten random samples of same textured image d19 with feature patterns

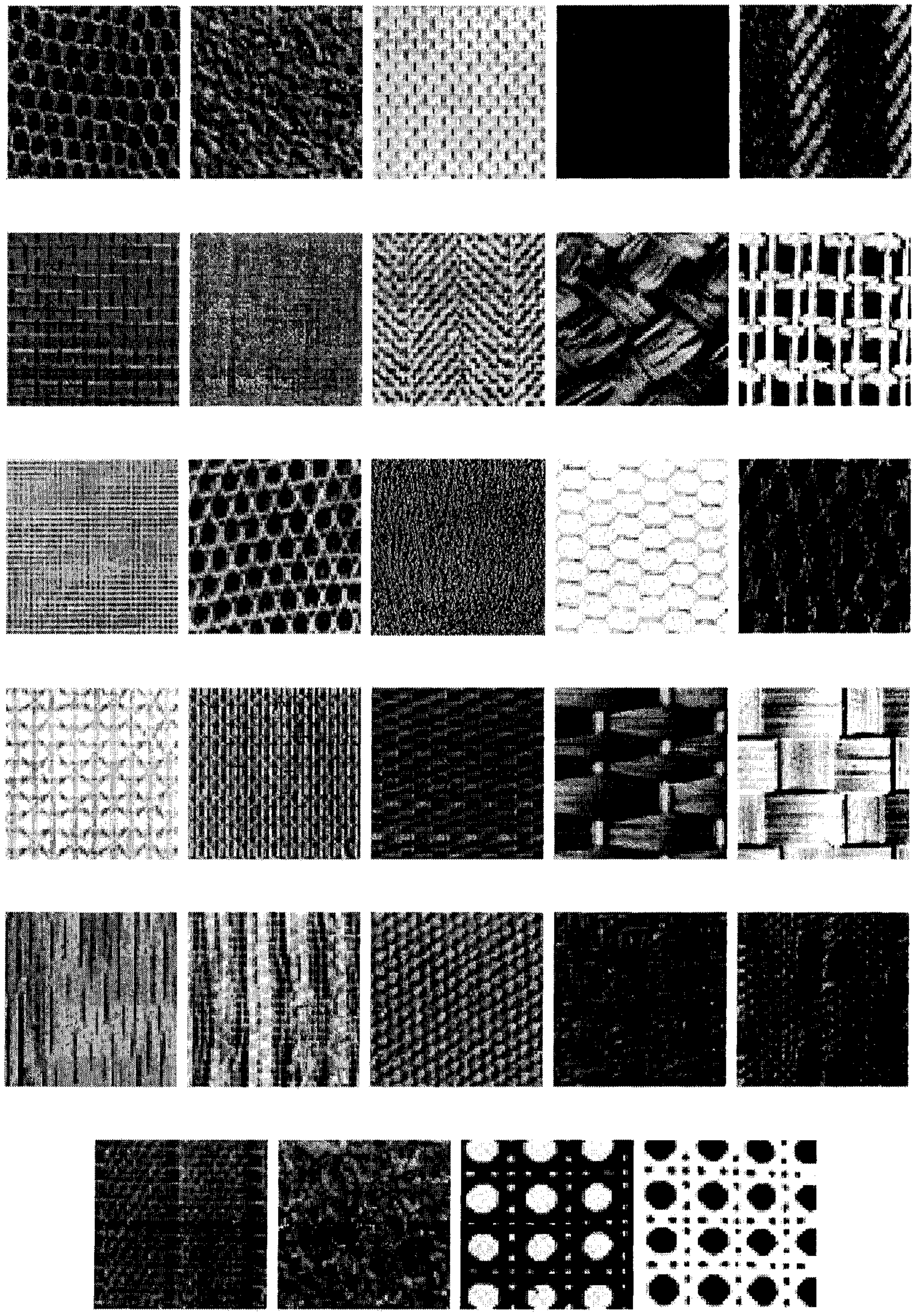


Figure B.12: Occurrence of d19 feature pattern within other 29 textured images

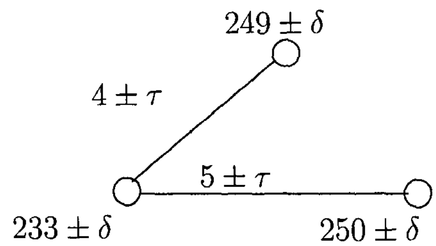


Figure B.13: One common frequent feature pattern of the texture class d34

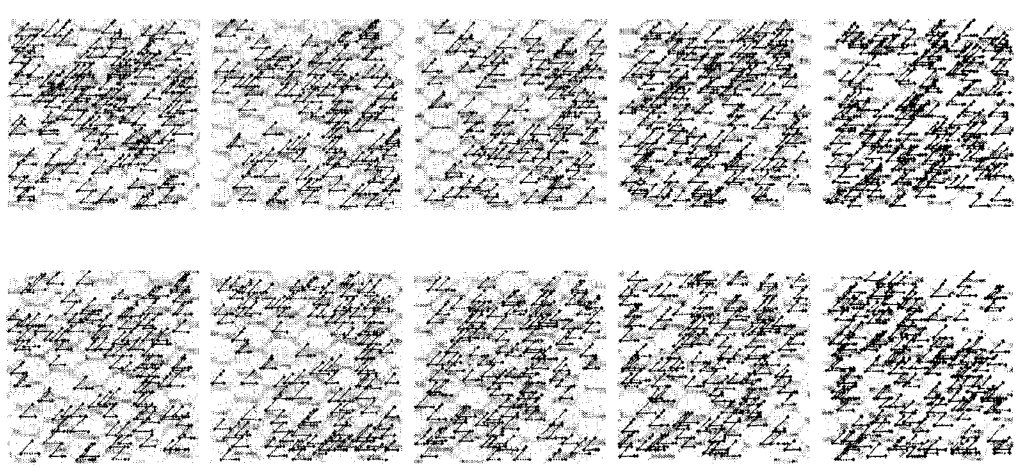


Figure B.14: Ten random samples of same textured image d34 with feature patterns

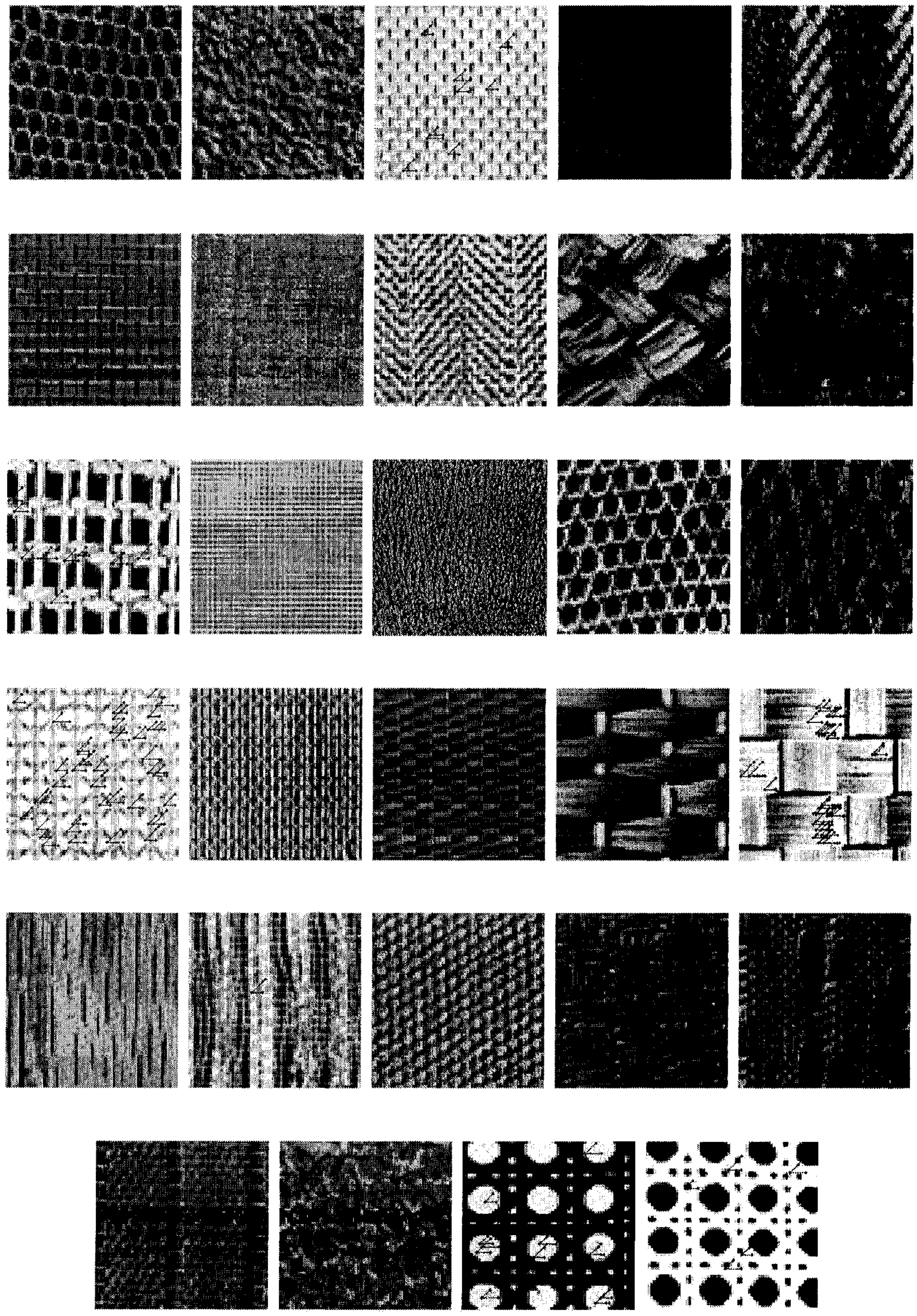


Figure B.15: Occurrence of d34 feature pattern within other 29 textured images

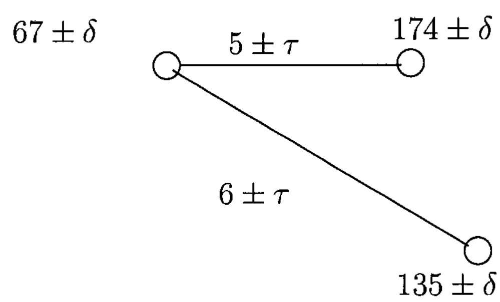


Figure B.16: One common frequent feature pattern of the texture class d35

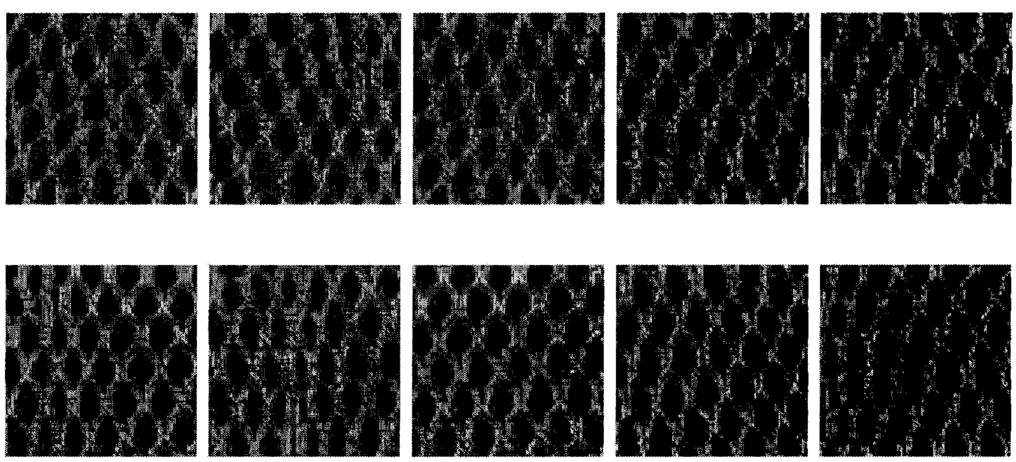


Figure B.17: Ten random samples of same textured image d35 with feature patterns

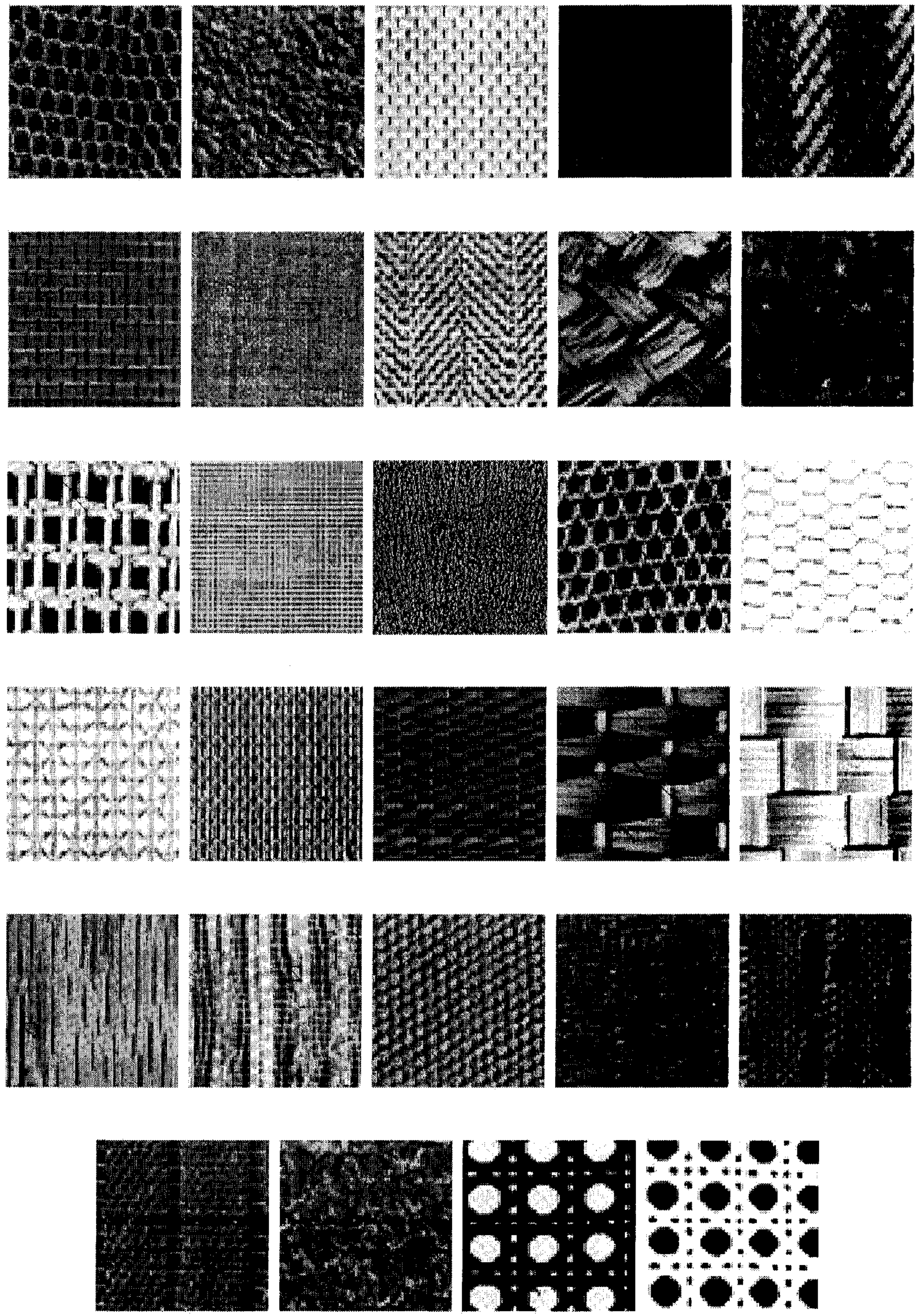
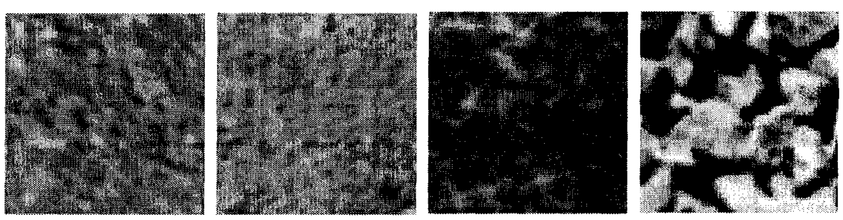


Figure B.18: Occurrence of d35 feature pattern within other 29 textured images

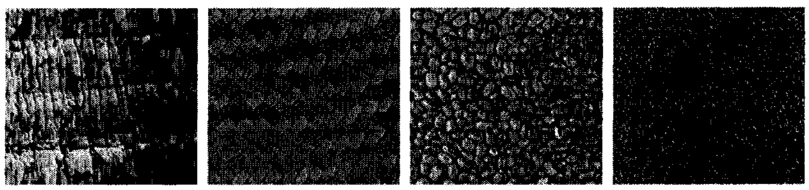
Appendix C

Textures of Meastex and Vistex databases

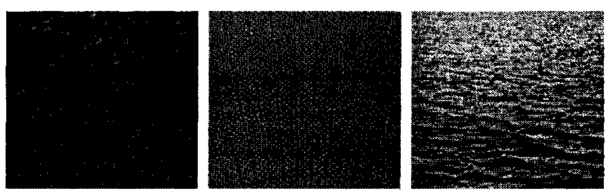


(a) Asphalt (b) Concrete (c) Grass (d) Rock

Figure C.1: Samples of Meastex database



(a) Bark (b) Fabric (c) Food (d) Metal



(e) Sand (f) Tile (g) Water

Figure C.2: Samples of Vistex database