

**CLUSTER CODING WITH MODIFIED FLOOD
FILL ALGORITHM FOR TEXTURE
SEGMENTATION**

KHOO HEE KOOI

UNIVERSITI SAINS MALAYSIA

2009

**CLUSTER CODING WITH MODIFIED FLOOD
FILL ALGORITHM FOR TEXTURE
SEGMENTATION**

by

KHOO HEE KOOI

**Thesis submitted in fulfillment of the
requirements for the degree of
Master of Science**

July 2009

ACKNOWLEDGEMENTS

I would like to specially thank my supervisor, Dr. Ong Hong Choon for sharing his ideas and his guidance. He has given me a lot of helpful guidelines for my research experiments. He has also provided me plenty of resources which are very helpful for the study. Secondly I am also grateful to my field supervisor, Mr. Wong Ya Ping for his help in solving the computer vision problems. He had shared his expertise in both programming techniques and imaging methodology, especially in visualization and image data transform.

This study is funded by the Malaysian Ministry of Science, Technology and Innovation (MOSTI) under the Grant Number 305/PMATHS/613122 of eScience Fund. The funding allowed me to attend and present my findings in the proceedings on the International Conference of the 5th Computer Graphics, Imaging and Visualization (CGIV) held in Penang, Malaysia.

I would also like to thank to my senior, Mr. Ang Sau Loong for taking me to the engineering campus library to borrow books which has helped in my research work. He also assisted me by his sharing on clustering algorithm with me, so that I have a better picture of the processes involved in pattern recognition.

Finally, I would like to express appreciation to my family and friends, for supporting me along the way while completing my research work and in my master thesis writing.

TABLE OF CONTENTS

ACKNOWLEDGEMENTS	ii
TABLE OF CONTENTS	iii
LIST OF TABLES	vi
LIST OF FIGURES	vii
LIST OF PUBLICATIONS	viii
LIST OF ABBREVIATIONS	ix
ABSTRAK	x
ABSTRACT	xi

CHAPTER 1 – INTRODUCTION

1.1	An Introduction to Digital Image Processing	1
1.1.1	Texture Segmentation	4
1.1.2	The Applications of Texture Segmentation	7
1.2	Objectives of Study	8
1.3	Format of Thesis	9

CHAPTER 2 – TEXTURE SEGMENTATION TECHNIQUES

2.1	Texture Feature Extraction	11
2.2	Image Enhancement	16
2.3	Feature Reduction	18
2.4	Texture Feature Selection	19
2.5	Image Smoothing	22

CHAPTER 3 – CLUSTER CODING WITH MODIFIED FLOOD FILL

ALGORITHM

3.1	Grey Level Co-occurrence Probabilities (GLCP) for Feature Extraction	23
3.2	Analysis of Texture Features	27
3.3	Feature Normalization	31
3.4	Feature Enhancement using Histogram Equalization	32
3.5	Semi-supervised Feature Selection	
3.5.1	Cluster Coding Algorithm	34
3.5.2	Feature Smoothing	37
3.5.3	Cluster Merging using Modified Flood Fill Algorithm	39
3.5.4	Segmentation System	43

CHAPTER 4 – RESULTS AND DISCUSSIONS

4.1	Experimental Setup	
4.1.1	Parameter Configuration	44
4.1.2	Test Images	45
4.1.3	Equipments and Software Used	47
4.2	Hypothesis of Result	48
4.3	Experimental Results	
4.3.1	The Effects of the Enhanced Feature Domain	49
4.3.2	Segmentation Outputs	53
4.4	Comparative Performances	57
4.5	Discussion and Conclusion on Hypothesis Made	62
4.6	Two Real Life Applications of Cluster Coding with Modified Flood Fill	63

CHAPTER 5 – CONCLUSION

5.1	Summary of Results	65
5.2	The Strength of Cluster Coding with Modified Flood Fill Algorithm	
5.2.1	Advantages of Our Proposed Method	66
5.2.2	Limitations of Our Proposed Method	67
5.3	Future Work	68

REFERENCES	69
-------------------	----

APPENDICES

Appendix A	Table of Brodatz’s Texture Characteristics	73
Appendix B	Source Code for GLCP Feature Extraction	74
Appendix C	Source Code for Feature Normalization	78
Appendix D	Source Code for Histogram Equalization	81
Appendix E	Source Code for Cluster Coding	87
Appendix F	Source Code for Feature Smoothing	91
Appendix G	Source Code for Modified Flood Fill Algorithm	94

LIST OF TABLES

		Page
Table 3.1	Statistical features for GLCP extraction	26
Table 4.1	GLCP parameters configuration	44
Table 4.2	Specification of computer systems used	47
Table 4.3	The parameter set used for the segmentation of Figures 4.3A to 4.3E	53
Table 4.4	The accuracies and computation times of KIF (Clausi, 2002b) and CMF	57
Table 4.5	The accuracies and computation times of SOM (Martens, 2008) and CMF	58
Table 4.6	The parameter set used for the segmentation of real imagery	64

LIST OF FIGURES

		Page
Figure 1.1	The visible electromagnetic spectrum (The Electromagnetic Spectrum, 2008).	1
Figure 1.2	The basic framework for texture segmentation.	6
Figure 3.1	The flowchart of the GLCP algorithm for feature extraction.	25
Figure 3.2A	The 10-partite of texture image.	27
Figure 3.2B to 3.2L	The graphs of 1D GLCP statistical features from Figure 3.2A.	27-29
Figure 3.3	The flowchart of the HE process for feature enhancement.	33
Figure 3.4	The segments for cluster coding.	34
Figure 3.5	The flowchart of the cluster coding algorithm.	36
Figure 3.6	The mapping of the array memory structure.	37
Figure 3.7	The mechanism of the smoothening method.	38
Figure 3.8	The flowchart of the modified flood fill algorithm for cluster merging.	42
Figure 3.9	The summary for the whole segmentation system.	43
Figure 4.1	The mosaic structures used for testing.	45
Figure 4.2	The 2D feature domain and the enhanced 2D feature domain.	49-52
Figure 4.3	Texture mosaics and the corresponding segmentation results.	54
Figure 4.4	The bar chart for comparison of the accuracies of CMF and KIF.	60
Figure 4.5	The bar chart for comparison of the times of CMF and KIF.	60
Figure 4.6	The bar chart for comparison of the accuracies of CMF and SOM.	61
Figure 4.7	The Black Hills of Wyoming and South Dakota (Norre, 2008).	63
Figure 4.8	The mammogram image of a breast with malignant tumour (BreastCancer.org, 2009).	64

LIST OF PUBLICATIONS

1. Khoo H. K., Ong H. C., Wong Y. P. (2008). Image texture classification using combined grey level co-occurrence probabilities and support vector machines. *Proceedings of the 5th International Conference on Computer Graphics, Imaging and Visualization, CGIV'08*, p.180-184.
2. Hong-Choon Ong, Hee-Kooi Khoo & Ya-Ping Wong (2009). Improved image texture classification using grey level co-occurrence probabilities with support vector machines post-processing. *Proceedings of the 5th Asian Mathematical Conference, AMC'09* – Presented on 22nd – 26th June 2009 at Putra World Trade Centre, Kuala Lumpur and under review for publication.

LIST OF ABBREVIATIONS

1D (One Dimensional)
2D (Two Dimensional)
3D (Three Dimensional)
ACF (Autocorrelation Function)
AI (Artificial Intelligence)
ANN (Artificial Neural Network)
API (Application Programming Interface)
CMF (Cluster Coding with Modified Flood Fill)
CPU (Central Processing Unit)
CDF (Cumulative Distribution Function)
DWT (Discrete Wavelet Transform)
EM (Expectation-Maximization)
FT (Fourier Transform)
FFT (Fast Fourier Transform)
GLCM (Grey Level Co-occurrence Matrix)
GLCP (Grey Level Co-occurrence Probabilities)
GPU (Graphics Processing Unit)
HE (Histogram Equalization)
HMM (Hidden Markov Models)
ICA (Independent Components Analysis)
PCA (Principal Components Analysis)
RAM (Random-Access Memory)
SVs (Support Vectors)
SVM (Support Vector Machines)

KELOMPOK PENGKODAN DENGAN ALGORITMA ISI BANJIR TERUBAHSUAI UNTUK PENSEGMENAN TEKSTUR

ABSTRAK

Tekstur merujuk kepada sifat yang menggambarkan permukaan atau struktur sesuatu objek dan ditakrifkan sebagai terdiri daripada unsur-unsur yang saling berhubung. Fokus utama dalam kajian ini ialah pensegmenan tekstur dalam imej digit dua dimensi. Satu siri algoritma-algoritma kacukan diperkenalkan untuk penyenggaraan atau memperbaiki kaedah dari segi ketepatan klasifikasi dan masa pengiraan. Algoritma-algoritma ini dibahagikan kepada empat tahap; iaitu pengekstrakan sifat, penguatan sifat, pemilihan sifat, dan perlicinan sifat. Kaedah kebarangkalian kejadian bersama paras kelabu (GLCP) digunakan untuk mengekstrak sifat-sifat daripada imej-imej tekstur. Sifat-sifat dalam bentuk statistik boleh dikira berdasarkan GLCP yang dijanakan. Sifat-sifat tekstur didapatkan dengan menggunakan kombinasi sudut-sudut yang berlainan bagi menghasilkan sedikit invariants putaran. Bagi menguatkan sifat-sifat tekstur, penyamaan histogram (HE) diterapkan kepada sifat-sifat dalam bentuk statistik. Kelompok pengkodan dengan algoritma isi banjir terubahsuai diperkenalkan bagi pemilihan sifat-sifat tekstur untuk menangani masalah-masalah, seperti corak-corak yang tidak tentu, gangguan, dan unsur-unsur terkeluar yang berlaku dalam set domain yang diekstrakkan. Kesemua sistem pensegmenan tekstur ditulis dalam bahasa pengaturcaraan C++ dengan satu dimensi barisan struktur data bagi mempercepatkan pengiraan, terutamanya ketika mengendalikan algoritma-algoritma yang melibatkan struktur gelung. Album tekstur Brodatz digunakan dalam kajian ini untuk menguji keputusan. Dalam kajian ini, kelompok pengkodan dengan algoritma isi banjir terubahsuai menunjukkan kemajuan yang nyata berbanding dengan teknik-teknik lain dari segi ketepatan klasifikasi and masa pengiraan.

CLUSTER CODING WITH MODIFIED FLOOD FILL ALGORITHM FOR TEXTURE SEGMENTATION

ABSTRACT

Texture refers to properties that represent the surface or structure of an object and is defined as something consisting of mutually related elements. The main focus in this study is to do texture segmentation in two dimensional (2D) digital images. A series of hybrid algorithms to segment textures is proposed in order to maintain or improve the classification accuracy and computation time. The algorithms are divided into four stages; these are feature extraction, feature enhancement, feature selection, and feature smoothening. Grey level co-occurrence probabilities (GLCP) method is being used to extract features from texture images. Statistical features can be calculated based on the GLCP generated. The features are obtained by using a combination of different angles to give some rotational invariance. To enhance the features, histogram equalization (HE) is applied to the statistical features. Cluster coding with modified flood fill algorithm is proposed for feature selection to resolve the uncertain texture patterns, noise, and outliers occurring on the extracted feature domain. The whole texture segmentation system is written in C++ programming language with one dimensional (1D) array data structure for fast computation, especially when dealing with algorithms involving iterations. Brodatz texture album is used in this study to test out the result. In this study, the cluster coding with modified flood fill algorithm showed a significant improvement over other techniques in terms of classification accuracy and computation time.

CHAPTER 1

INTRODUCTION

1.1 An Introduction to Digital Image Processing

A digital image is a two dimensional (2D) finite set of picture elements (pixels) where each pixel is arranged to show a picture of something that the human eyes could see. Every pixel of an image is defined by coordinates and a set of tonal properties. The coordinates and the set of tonal properties of each pixel can be assigned either synthetically or captured by an optical device based on the visible electromagnetic spectrum, as shown in Figure 1.1 taken from The Electromagnetic Spectrum (2008).

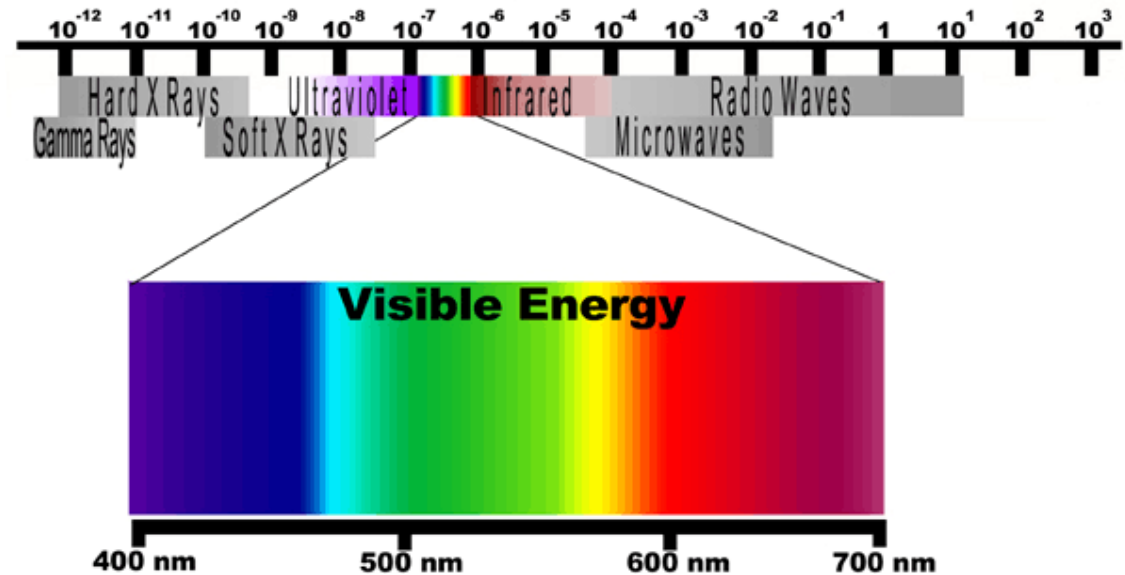


Figure 1.1: The visible electromagnetic spectrum (The Electromagnetic Spectrum, 2008).

In early 1920s, the first use of digital images was started in the newspaper industry. The invention of the Bartlane cable pictures transmission system created during that time was used to send the pictures through the Atlantic Ocean between London and New York. The transmission system coded the pictures in order to be sent through cables. The codes were then reproduced using a telegraph printer. The first coded pictures have limited luminance and there were some faulty appearances on the pictures. Besides, the transmission takes three hours to complete and more efficient equipments were required in order to reduce the time (Gonzalez & Woods, 2006).

The development of digital image processing was still not possible in the 1930s and 1940s because the essential components such as, the large data storage and high computational equipments were not available during this period of time (Gonzalez & Woods, 2006). In the 1940s, early digital computers were invented. One of the digital computers being established during that time was the Electronics Numerical Integrator and Computer (ENIAC). ENIAC was a computing device and programmable in decimal numerical system. However, most of these digital computers were used only for military purpose, but not for digital image processing (Rojas & Hashagen, 2002).

Digital image processing involves processes which simplify a digital image into a couple of versions of pictures for the human to understand more about the digital image shown. The foundation of digital image processing was established in the 1960s, at University of Maryland (Rosenfeld, 1969). The processes consist of segmentation, filtering, restoration, enhancement, etc. However, the processing of digital images was costly due to the limitations of the computing equipments during that time. The

equipments were also expensive and so the digital image processing activities could only be practically performed by certain institutes funded by the government or an industrial company.

Formula Translation or FORTRAN, the first programming language was developed by IBM in 1954. This programming language is a scientific oriented language, which is used for solving complex mathematical formula. FORTRAN was found to be difficult to program and is then being replaced by C in 1970s and later C++ in 1980s (Capron & Johnson, 2004). The C programming language was created by Dennis Ritchie and then the expanded version of C, the C++ was developed by Bjarne Stroustrup at Bell Laboratories (Deitel & Deitel, 2007). Most of the application programming interfaces (API) involving graphics are written in the standard C and C++ programming language, such as the OpenGL toolkits developed by the Silicon Graphics Inc. in 1992 (OpenGL – The Industry Standard for High Performance Graphics, 2008).

In the year 2000 onwards, digital image processing had become common with the advance technology of the random-access memory (RAM), central processing unit (CPU) processor (Intel® Processors, 2009), and graphics processing unit (GPU) processor (Nvidia – World Leader in Visual Computing Technologies, 2009). The work of digital image processing requires a wide range of mathematical equations and algorithms ranging from as simple as intensity threshold to the challenging ones, such as pattern recognition. In digital images, patterns which we want to recognise involve textures (Gonzalez & Woods, 2006).

Today, the performance of computing hardware has rapidly grown. Thus much more complex algorithms can be integrated into digital image processing. The research challenges now aim for the non-supervised, high performance, efficient, and reliable methods for applications in real life environment (Zhang et al., 2008a). The application of digital image processing with the knowledge base involving artificial intelligence (AI) and machine vision have become important elements to produce high quality, fast processing, and multi-purpose electronics devices to assist mankind.

1.1.1 Texture Segmentation

A texture is defined as a pattern and is represented on the surface or structure of an object. The arrangement of the texture patterns can be random in nature or having several kinds of geometrical transformations, such as translation, scaling or rotation. In most captured real life digital images, the existence of texture is inevitable.

Texture segmentation refers to the separation of the image into regions, each of which is occupied by a single texture type. Segmentation process which involves texture discrimination is complicated. There are intermediate tasks that are needed to be done. These include finding several characteristics of a texture that are needed to be identified, retrieving the spatial information about the correlation between neighbouring pixels and also the procedure for noise reduction strategy. The high level of computation for texture segmentation is required in order to achieve the reasonable and accurate results.

In human vision, a texture can be described qualitatively in several different ways. These terms include randomness, stationary, coarseness, periodicity, roughness, brightness and so on. These terms are subjective and lack measurements to quantitatively describe a texture, and thus further analysis of texture is not possible to proceed. In order to explain texture in a quantitative way, the distribution of a texture can be described in the form of statistics, differential equations, geometry, and fractal analysis.

For the past few decades until now, many researchers from various fields have been trying to develop algorithms to do texture discrimination. Alfréd Haar created the first discrete wavelet transform (DWT) in Haar (1910) which led to the development of the fast Fourier transform (FFT) and the other forms of DWT to detect periodic signals. Haralick had developed grey level co-occurrence probabilities (GLCP) with statistical features to describe textures (Haralick et al., 1973). The hidden Markov models (HMM) are another statistical methods used for pattern recognition (Rabiner, 1989).

The general framework of texture segmentation is divided into two important stages, which are feature extraction and feature selection. Feature extraction is a process of transforming a texture into a feature domain based on the intensity distribution on a digital image. On the other hand, feature selection is a process of integrating a set of conditional statements and routines, so that the computer can logically decide which pixels belong to which texture. Therefore, feature selection is the subject of pattern recognition.

Figure 1.2 summarizes the sequence involved in the process of texture segmentation. The segmentation process starts with image acquisition to retrieve the partial information carried by the test image and then followed by image transformation in order to select textures based on the extracted features. Feature observation helps in searching the appropriate texture features and therefore it can be part of the feature selection stage. Feature observation consists of three types of methods, which are unsupervised, semi-supervised, and supervised method.

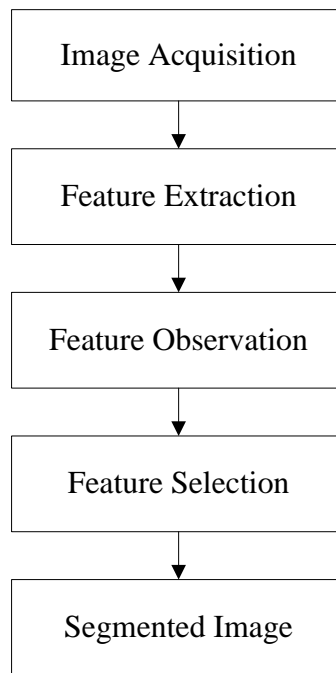


Figure 1.2: The basic framework for texture segmentation

1.1.2 The Applications of Texture Segmentation

Texture segmentation techniques are applied to several sectors. These sectors involve machine vision, medical imaging, biometrics, satellite applications, food science, geological studies, crime scene investigation, and so on.

The texture segmentation techniques are important tools to assist in medical surgery. For example, to classify the tumour of breast cancer in Karahaliou et al. (2007) based on mammography imagery, to detect abnormalities in patients using Magnetic Resonance Imaging (MRI) imagery (Zhang et al., 2008b), and precisely segmenting human brain images on Computed Tomography (CT) imagery for visualization during surgery (Tong et al., 2008).

In food science, texture segmentation techniques aid on improving the quality of food and decrease the rate of food poisoning cases. For instance, a non-supervised method of texture segmentation is proposed to estimate the content of Intramuscular Fat (IMF) in beef on Du et al. (2008) and thus improving the meat quality.

In remote sensing, texture segmentation is used to analyze satellite Synthetic Aperture Radar (SAR) imagery for multiple purposes. The analyzed satellite imagery might be used to monitor flood situations (Seiler et al., 2008), large scale construction planning, archeological research, weather forecasting, geological study etc. Texture segmentation is also applied in the study of global warming by detecting and estimating the size of the icebergs at the North Pole from time to time (Clausi, 2002a).

Another breakthrough in the advancement of security is in biometric authentication. One of the biometric measures is in using the human eye as an identity of a person. The human iris contains texture patterns which are unique in nature and thus texture classification techniques can be applied on this iris imagery (Bachoo & Tapamo, 2005).

1.2 Objectives of Study

There are four major goals in this thesis which are as follows:-

- a. To study how statistical texture descriptions can be used for image segmentation.
- b. To develop algorithms for feature extraction based on texture images.
- c. To develop algorithms for texture segmentation based on texture features analyzed.
- d. To improve the performance of the segmentation system in terms of accuracy and time.

1.3 Format of Thesis

Chapter 1 is an introduction on the digital image processing and texture segmentation. This chapter presents the achievements and events which lead to the higher level of digital image processing techniques. This chapter also introduces the existing texture segmentation techniques and the applications of the texture segmentation in solving real life problems. The goals for this study in improving the quality and reliability of existing texture segmentation techniques are justified.

In Chapter 2, we shall discuss about the existing texture segmentation techniques and identify the limitations of some of these techniques. The discussion is divided into five progressive stages, which are feature extraction (Section 2.1), feature enhancement (Section 2.2), feature selection (Section 2.3), feature reduction (Section 2.4), and feature smoothening (Section 2.5).

The methodology in Chapter 3 is divided into three parts. Firstly we will describe the GLCP method for feature extraction. Then we will present the observations of textures picked from Brodatz imagery album and the intermediate data transformation techniques used. The last section will described the proposed method, namely cluster coding with modified flood fill (CMF) algorithm based on the findings from the second part of the chapter.

Chapter 4 states the necessary preparations and precautions that need to be taken note before doing the texture segmentation tasks. The details involve the parameters configuration, the list of test images used, making the hypothesis of the expected result (Section 4.2), and the technical description on the equipments used. The results of the CMF as compared to other existing methods are shown in Chapter 4. Furthermore, this chapter also shows the CMF being applied to two real life problems.

The final chapter will conclude on the summary of the result. The strength of the CMF is identified and reported (Section 5.2). We also suggest some ways for further improvements based on the limitations of the proposed method in the future.

CHAPTER 2

TEXTURE SEGMENTATION TECHNIQUES

2.1 Texture Feature Extraction

There are several mathematical models which are used to extract texture features from an image. This section discusses the five main feature transformations, namely autocorrelation function (ACF), Gabor filter, discrete wavelet transform (DWT), hidden Markov models (HMM), and grey level co-occurrence probabilities (GLCP). Besides, some combined methods using different models are also developed to increase the robustness of texture segmentation system. We will briefly discuss two hybrid methods; these are Gabor wavelet and wavelet-domain HMM.

Gabor filter is one of the texture descriptors based on the Fourier transformation (FT). The test image is first given a FT to generate a 2D sinusoidal signal. For texture recognition, the Gabor filter bank contains a list of sub-bands of different signals generated by different textures. After the sub-bands are determined, the 2D signal of test image will multiply one of the chosen sub-bands and yield only the frequencies that match the sub-band. The product is then transformed back by taking the inverse FT and this leaves only the location of the texture feature which matches the signal. The process continues with each possible sub-band and produces the locations where the same signals occur (Petrou & Sevilla, 2006).

Gabor filter is useful in adapting sinusoidal signals whereby it can be decomposed into a weighted sum of sinusoidal signals. Thus Gabor filter is suitable to decompose textural information. Experiments done by Clausi & Deng (2005) stated that the Gabor filter can well recognize low and medium frequencies, but it produces inconsistent measurements for high frequencies due to the noise in the signal. The feature domain generated by Gabor filters are not distinctive enough for high frequencies and thus could affect the segmentation result (Hammouda & Jernigan, 2000).

Texture pattern can also be modeled as a transitional system using hidden Markov models (HMM). A Markov model assumes a texture pattern have a finite number of states and times. Each probability of a state is determined by the previous probability of the state. Three issues can arise from HMM observations; these are evaluation, decoding, and learning. HMM evaluation is to compare the probabilities of different models which best describe the texture feature. HMM decoding is to decompose and provide an estimated basis of texture patterns based on the HMM observations. The HMM learning searches for which model parameters best describe the texture pattern (Sonka et al., 2007).

The HMM can generate a consistent measurement for texture patterns based on the best probabilities. However, one or more observations which produce undesired probabilities could generate disorderly sequences due to the noisy pattern in the test image (Sonka et al., 2007).

Autocorrelation function (ACF) is another method to describe texture patterns. The function helps to search for repeated patterns in a periodic signal. The function also identifies the missing basis of texture patterns hidden under the noisy patterns. In using the ACF, the mean of each image is adjusted before applying the general formula. Thus we are actually computing the normalized auto-covariance function. One can characterize a texture pattern by inferring the periodicity of the pattern (Petrou & Sevilla, 2006).

The ACF feature is well demonstrated and distinctive between textures on a three dimensional (3D) graph. In feature selection, inferring the periodicity of a texture feature is done by observing several threshold points of the auto-covariance function and then counting the number of peaks for each threshold in a fixed variation. This may result in a random fluctuation and texture segmentation may fail because of two issues; there is not enough information by taking only one dimensional (1D) threshold to compare and an appropriate set of standard deviation of the distances between peaks are needed to know when the periodicity end. For example, the lagged product estimator and time series estimator are proposed to select ACF feature in Broerson (2005). But to appropriately characterize the texture pattern by its periodicity is still an active area of research.

Instead of using the Gabor filter for feature extraction, wavelet is well known today as a flexible tool to analyze texture. Wavelet is a function whereby the basic function, namely the mother wavelet is being scaled and translated in order to span the spatial frequency domain. The DWT is done by the product of a corresponding signal generated by a pattern and the complex conjugate of the wavelet, and then integrating

over all the distance points conveyed by the signal. In texture analysis, a tree data structure namely a packet wavelet expansion is used to split the signal into smaller packets and expand a chosen band at each level of resolution (Petrou & Sevilla, 2006).

The DWT can become rotation invariant in texture analysis by taking the logarithm of the frequency sub-band. But this will damage the frequency components of the corresponding signal generated by a texture pattern and thus may obtain inaccurate result (Khouzani & Zadeh, 2005).

Grey level co-occurrence probabilities (GLCP) method is a non-parametric solution whereby the textures are described in a discrete domain (Petrou & Sevilla, 2006). GLCP statistics are used to preserve the spatial characteristics of a texture. The selection of certain texture is possible as it is based on the statistical features. The best statistical features that are used for analysis are entropy, contrast, and correlation (Clausi, 2002a). However, further analysis in Jobanputra & Clausi (2006) shows that correlation is not suitable for texture segmentation. GLCP statistics can also be used to discriminate between two different textures. Boundaries can be created from the shift on statistical feature while moving from one texture to another (Jobanputra & Clausi, 2006). Clausi & Zhao (2002) also proposed grey level co-occurrence linked-list (GLCLL) structure and grey level co-occurrence hybrid histogram (GLCHH) structure in Clausi & Zhao (2003) to this non-parametric solution for storing purpose in order to speed up the computational time for GLCP feature extraction.

Wavelet-domain hidden Markov models (HMM) can be described as a finite state machine in the wavelet domain. The hidden Markov tree (HMT) can identify the characteristics of the joint probabilities of DWT by capturing the scale dependencies of wavelet co-efficient via Markov chains (Fan & Xia, 2003). Since this method is based on the wavelet domain, the disorder sequence of signal generated by the noise in the texture patterns may weaken the cross-correlation between DWT sub-bands (Fan & Xia, 2003). Ming et al. (2008) proposed the wavelet hidden-class-label Markov random field to suppress the specks of noise, but there are some small blobs of noises still appearing in the results of the segmented imagery.

Gabor wavelet is another hybrid method to analyze texture patterns. The Gabor wavelets or the Gabor elementary functions (GEF) are Gaussians modulated by complex sinusoids. GEF could achieve the minimum frequency bandwidth product and generate a non-orthogonal basis for patterns being analyzed (Manthalkar & Biswas, 2002). Since Gabor wavelets are not in orthogonal decomposition, this means that a wavelet transform based on the Gabor wavelet is redundant. Besides, the computation of the expansion set of co-efficient also becomes difficult due to the non-orthogonal property of the Gabor wavelets (Ferrari et al., 2001).

2.2 Image Enhancement

There are two major developments of image enhancement, which are image enhancing in the spatial domain and image enhancing in the frequency domain. Image enhancing in the spatial domain manipulates the enhancement directly based on the pixels in the image itself. Image enhancing in the frequency domain is based on the normalized feature distribution (Gonzalez & Woods, 2006). Since we are focus on texture pattern, so in this section it is necessary to discussion the image enhancement in the frequency domain.

The main objective of the texture feature enhancement is to obtain a higher separability of the extracted features after using the feature extraction methods as stated in Section 2.1. Feature enhancement is important to improve the segmentation accuracy, because it could reveal the texture patterns which are buried by the noise that occurs in the frequency domain. Indirectly, it reveals that the patterns could enhance the visibility of the boundaries at the edges between the textures.

Histogram equalization (HE) is a transformation function based on the intensity histogram to improve the contrast of an image. HE rescales the range of intensity in feature domain to produce pixel values which are more uniformly distributed (Martens et al., 2008). The implementation of HE starts by constructing an intensity histogram from the spatial domain. Then, a cumulative distribution function (CDF) for each pixel is computed from the intensity histogram. The normalized version of CDF is corresponds to the HE.

Du et al. (2008) used the bilateral filtering to remove noise that appears on the beef images captured by the CCD (Charge Couple Device) camera. Bilateral filtering was proposed by Tomasi & Manduchi (1998) as a heuristic tool for noise removal in beef images. The method configures a range of frequency domain to match the similarity of the pixels involved, and then estimates the shape of a texture with geometric proximity.

Arslan & Grigoryan (2007) proposed a paired transform in the discrete Fourier transform (DFT) signal to enhance medical images. This transformation first compute the paired splitting signals based on one dimensional (1D) DFT, then perform the weighted paired splitting signals to improve the contrast of an image from the splitting signals. The weights in paired transform is still arbitrary, an appropriate set of weights needed be justify semantically in order to obtain a uniformed enhancement.

Alpha rooting is an algorithm used to enhance images in orthogonal transform signals. Orthogonal transform methods include the DFT, wavelets, Gabor filter etc. This algorithm only focus on the high frequency signals and it may result in a local instead of overall contrast enhancement (Panetta et al., 2008).

2.3 Feature Reduction

Feature reduction is necessary when the number of texture features is large. A graphical representation of the features is not available for data observations with more than three dimensions. Feature reduction chooses only a subset of the large features to measure texture patterns. However, the selection of the best subset of features is an expensive search. This problem can be resolved using optimization techniques explained in Chen & Wang (2005).

Principle components analysis (PCA) is one of the techniques used to reduce the dimensionality of texture features. PCA can be defined as the orthogonal projection of the data onto a lower dimensional space (Bishop, 2006). The process involves identifying the main texture patterns by highlighting their similarities and differences. Then, PCA provides an optimal texture features in order to help reduce the dimensionality of large features in texture analysis.

PCA can reduce the dimensional space and improve the computational time, but Clausi & Deng (2005) do not recommended PCA feature reduction. This is because if we use the PCA transform for a given feature set, the quality of the segmentation is expected to remain the same or become poorer.

2.4 Texture Feature Selection

The texture feature selection stage is the most important part of the texture segmentation process because it determines which pixels belong to which texture of an image. The use of parameters of a chosen method is crucial to the output of the result.

K-means algorithm has become a popular clustering method which is used for pattern recognition. Given the number of clusters, K , the algorithm will start at a random K number of centres. Then, each of the centres will group the features using the closest distances or Euclidean distance measures. The locations of the features with the same cluster will determine the new centre for each cluster. The process will then repeat until the centre of each texture class remains the same.

K-means algorithm assumes that all clusters are in spherical shape, but it may return inappropriate result for non-spherical clusters (Clausi, 2002b). In the real life 2D feature set is not always in spherical shape and normally the number of classes is unknown.

Support vector machines (SVM) algorithm is a slow but highly accurate clustering method. The SVM training algorithm was introduced by Boser et al. (1992). The purpose of SVM is to map feature vectors into a higher dimensional feature space, and then creating a separating hyperplane with maximum margin to group the features. Support vectors (SVs) contain highlighted pixels that help to create the margins or boundaries in an image. The higher dimensional space is defined by a kernel function.

Some of the popular kernels are shown in Schölkopf & Smola (2002). A combined grey level co-occurrence probabilities (GLCP) and SVM technique has been proposed for two class segmentation with significant result in Khoo et al. (2008).

Expectation-maximization (EM) algorithm is a statistical estimation algorithm used for finding maximum likelihood and estimates of parameters in probabilistic models. The parameters involve means, variances, and weights (Tong et al., 2008). The EM algorithm starts by initializing the parameters to compute the joint probability for each cluster. The algorithm then iterates to re-estimate the parameters and maximize the probability for each cluster until the set of convergence values of probabilities are obtained (Bishop, 2006). The convergence probabilities are only dependent upon the statistical parameters, so we must carefully choose these parameters, especially the parameters for texture patterns (Diplaros et al., 2007).

The self-organizing map (SOM) is an unsupervised single layer artificial neural network (ANN). In the SOM training environment, a digital image is mapped as a grid. A set of neurons will be placed at random grid points where each neuron is stored as a cluster centre (Chen & Wang, 2005). SOM clusters regions which have similar pattern and separates the dissimilar patterns based on a general distance function (Martens et al., 2008). The SOM learning process is similar to the K-means clustering where it iterates until each of the cluster centre converges to the centre of the possible texture patterns.

The advantage of SOM is that the more number of neurons are placed in the grid, the higher classification result will be obtained. However, if the numbers of neurons are too large, SOM may end up with over classification (Martens et al., 2008). On the other hand, the numbers of neurons required is unknown. Furthermore, the classification using SOM may fail at the local minimum in training (Abe, 2005).

We have briefly discussed the principle component analysis (PCA) in Section 2.3 which is used to resolve the large dimensional of texture features. PCA could provide partial information on the statistical distribution of textures. However, PCA fail to discriminate partial information which is non-correlated. Independent component analysis (ICA) is introduced to discriminate non-correlate frequencies. Besides, ICA could produce independent statistical distribution for each non-correlated frequency (Chen & Wang, 2006).

2.5 Image Smoothing

Image smoothing techniques are used to suppress noise that occurred on the image. The tradition of the technique reconsiders each pixel by taking the average of the brightness values in some neighbourhood pixels. The disadvantage of these techniques is that smoothing could cause the problem of blurring edges in the image. Moreover, pre-processing using local image smoothing can effectively remove noise appearing in the form of thin stripes, but does not work if the noise is in the form of thick stripes (Sonka et al, 2007).

For texture segmentation purposes, an alternative technique of smoothing in which edges are preserved, such as the weighted grey level co-occurrence probabilities (weighted-GLCP) are used in Jobanputra & Clausi (2006). However, this post-processing method may violate the statistical distribution collected from the grey level co-occurrence matrix (GLCM). The weighted-GLCP method has shown improvements in Jobanputra & Clausi (2006), but is still not accurate due to the poor feature extraction for GLCP.

CHAPTER 3

CLUSTER CODING WITH MODIFIED FLOOD FILL ALGORITHM

3.1 Grey Level Co-occurrence Probabilities (GLCP) for Feature Extraction

Grey level co-occurrence matrix (GLCM) is a discrete function that represents joint probabilities, C_{ij} , of different sets of pixels having different grey levels. In practice, the co-occurrence matrix can be constructed by configuring a displacement vector between two pixels, (δ_x, δ_y) . The distance between two relational pixels is normally set to become 1 for simplicity. The common angle used is either 0° , 45° , 90° or 135° . The combination of angles can also be used for isotropic texture image (Jobanputra & Clausi, 2006).

The co-occurrence matrix is then divided by the total sum of all entries that give rise to the joint probability density function as defined by,

$$C_{ij} = \frac{F_{ij}}{\sum_{i,j=0}^{G-1} F_{ij}} \quad (3.1)$$

where F_{ij} represents the frequency of occurrence between grey levels, i and j (Haralick et al., 1973). The grey level quantization, G , recommended setting is 24 in Clausi (2002a). However, an adequate grey level setting is 64 in Jobanputra & Clausi (2006).

In GLCM, the various types of textures could carry different correlation information. The selection of statistical features can be determined through the frequencies of the number of textures, N in a synthetic N -partite image. In order for GLCP statistics to properly detect certain natural texture, a reasonable window size has to be set to extract the features. More analysis on co-occurrence matrix can be found in Carr & Miranda (1998). Some of the statistical features used in this study are defined and shown in Table 3.1.

Given a test image, the grey level intensity of each pixel on the image is calculated by taking the average of the primitive colour intensities as shown below,

$$\text{Grey Level Intensity, } I_i = \frac{r_i + g_i + b_i}{3} \quad (3.2)$$

where r , g and b are red, green and blue colour intensities respectively (Gonzalez & Woods, 2006). The subscript i represents the location of each pixel on the image. Equation 3.2 extracts a greyscale from the coloured image with ranges from the black colour as the lowest intensity, to the grey tones and ending with the white colour as the highest intensity. On the other hand, the primitive colour intensities of each pixel is set to the same value which is equal to I_i .

Given a window size, $M \times N$ and the displacement vectors, (δ_x, δ_y) , the GLCP statistical features of the test image in this study is computed using the GLCP algorithm as shown in Figure 3.1.